

PROTOTIPO DE CRM, USANDO SALESFORCE, PARA LA CREACIÓN DE UN GESTOR DE PRÁCTICAS

CRM PROTOTYPE, USING SALESFORCE, FOR THE CREATION OF AN PRACTICE MANAGER



TRABAJO FIN DE MÁSTER
CURSO 2020-2021

AUTOR
JAVIER LÓPEZ QUESADA

DIRECTOR
ANTONIO SARASA CABEZUELO

MÁSTER EN INGENIERÍA INFORMÁTICA
FACULTAD DE INFORMÁTICA
UNIVERSIDAD COMPLUTENSE DE MADRID

PROTOTIPO DE CRM, USANDO SALESFORCE, PARA
LA CREACIÓN DE UN GESTOR DE PRÁCTICAS
CRM PROTOTYPE, USING SALESFORCE, FOR THE
CREATION OF AN INTERNSHIP MANAGER

TRABAJO DE FIN DE MÁSTER EN INGENIERÍA INFORMÁTICA
DEPARTAMENTO DE SISTEMAS INFORMÁTICOS Y COMPUTACIÓN

AUTOR

JAVIER LÓPEZ QUESADA

DIRECTOR

ANTONIO SARASA CABEZUELO

CONVOCATORIA: SEPTIEMBRE 2021

CALIFICACIÓN: 8

MÁSTER EN INGENIERÍA INFORMÁTICA
FACULTAD DE INFORMÁTICA
UNIVERSIDAD COMPLUTENSE DE MADRID

07 DE SEPTIEMBRE DE 2021

AGRADECIMIENTOS

A mi madre y a mi padre, ya que sin ellos nada de esto sería posible. A mi familia, por apoyarme y animarme siempre, incluso cuando las cosas se torcían. A mi novia, por animarme y, sobre todo, aguantarme durante el desarrollo de este proyecto. A mis amigos de la infancia, que pese a la distancia siempre me han apoyado y animado en esta aventura. A mis amigos y compañeros de la universidad, en especial a Pedro y Álvaro, por ayudarme siempre que lo he necesitado y hacerme el máster más fácil y llevadero. A todo el profesorado, por formarme a nivel académico y personal en la informática. A mis compañeros de trabajo, en especial a Hugo, Eva y Mar, por enseñarme el apasionante mundo de Salesforce y ayudarme en todo momento. A Daniel Ángel Chaver Martínez, Vicedecano de Relaciones Externas e Investigación, por concederme una entrevista que me despejó muchas dudas. A todas las personas que forman o han formado parte de mi vida, gracias.

RESUMEN

El proceso de creación de la mayoría de aplicaciones suele ser costoso, perdiéndose mucho tiempo en tareas que, pese a ser muy comunes, la gran mayoría de lenguajes o herramientas no facilitan su implementación. Además, hay que tener en cuenta multitud de factores como la seguridad o el posterior soporte de la aplicación que obligan a que exista detrás un equipo con multitud de perfiles. Sin embargo, rara vez se piensa en los sistemas de gestión de relaciones con los clientes (CRM) como una solución válida a este problema, pese a que estos aportan una rápida transformación digital de negocio mediante bots, automatizaciones y la inclusión de herramientas muy innovadoras en el sector como, por ejemplo, tableau (Business Intelligence) o blockchain (Data Mining).

El objetivo principal del presente proyecto es adaptar el concepto de CRM, más concretamente de Salesforce CRM, a una aplicación de gestión de prácticas universitarias, pudiendo ser un sustitutivo del gestor usado por la UCM. El motivo de esta adaptación es debido a que existen conceptos y procedimientos en los modelos de negocio que se asemejan mucho a lo que sería un proceso estándar de gestión de prácticas en empresas, por lo que en este proyecto se han reutilizado procesos implícitos en el propio Salesforce y customizado otros que se alejan más del objetivo final de un CRM.

Para cumplir el objetivo anterior, se ha implementado una aplicación web que permite tanto a un alumno como a una empresa acceder a Salesforce sin necesidad de conocer nada del CRM. A su vez también permite al coordinador de prácticas administrar con mayor facilidad todos los procesos ligados a la gestión de prácticas. Entre las funcionalidades que se han implementado destacan la firma de contratos, la concesión de documentos oficiales y las alertas sobre empleos.

Palabras clave

Prácticas, CRM, Integración, Automatización, Salesforce, Angular, Administración de Relaciones con el Cliente

ABSTRACT

The process of creating most applications is usually costly, wasting a lot of time in tasks that, despite being very common, the vast majority of languages or tools do not facilitate their implementation. In addition, there are many factors to be taken into account, such as security or the subsequent support of the application, which require the existence of a team with a multitude of profiles. However, customer relationship management (CRM) systems are rarely thought of as a valid solution to this problem, despite the fact that they provide a rapid digital transformation of business through bots, automations and the inclusion of highly innovative tools in the sector such as, for example, tableau (Business Intelligence) or blockchain (Data Mining).

The main objective of the present project is to adapt the CRM concept, more specifically Salesforce CRM, to a university internship management application, being able to be a substitute for the manager used by UCM. The reason for this adaptation is due to the fact that there are concepts and procedures in the business models that are very similar to what would be a standard process of internship management in companies, so in this project we have reused processes implicit in Salesforce itself and customized others that are further away from the ultimate goal of a CRM.

To meet the above objective, a web application has been implemented that allows both a student and a company to access Salesforce without needing to know anything about CRM. At the same time, it also allows the internship coordinator to manage all the processes related to internship management more easily. Among the functionalities that have been implemented are the signing of contracts, the issuing of official documents and job alerts.

Keywords

Practices, CRM, Integration, Automation, Salesforce, Angular, Customer relationship management

ÍNDICE DE CONTENIDOS

Capítulo 1 -	Introducción.....	1
1.1	Motivación	1
1.2	Objetivos	2
1.3	Plan de trabajo	3
Capítulo 2 -	Estado de la cuestión.....	5
2.1	Sistemas de gestión de contenidos (CMS)	5
2.2	Planificación de recursos empresariales (ERP)	6
2.3	Gestión de las relaciones con los clientes (CRM)	7
2.3.1	Microsoft	8
2.3.2	Adobe	9
2.3.3	SAP	10
2.3.4	Oracle	10
2.3.5	Salesforce	11
2.4	Proyectos y artículos relacionados con Salesforce	13
Capítulo 3 -	Tecnologías empleadas.....	15
3.1	Tecnologías usadas para frontend	15
3.1.1	Angular	15
3.1.1.1	HTML5.....	15
3.1.1.2	TypeScript	15
3.1.1.3	SASS.....	16
3.1.1.4	Bootstrap	16
3.1.2	Salesforce	16
3.2	Tecnologías usadas para backend	16

3.3	Otras herramientas	17
3.3.1	Salesforce Developer Edition	17
3.3.2	Github	18
3.3.3	Visual Studio Code	18
3.3.4	Postman	18
3.3.5	Beeceptor	19
3.3.6	Complementos para Salesforce	20
	3.3.6.1 Salesforce AppExchange	20
	3.3.6.2 Salesforce Inspector	20
	3.3.6.3 Data Loader	21
	3.3.6.4 Funciones Einstein	21
3.3.7	Wix	21
3.3.8	Editores gráficos	22
Capítulo 4 -	Salesforce CRM.....	23
4.1	Conceptos básicos	23
4.1.1	Organización de elementos	24
	4.1.1.1 Estructura	24
	4.1.1.2 Administración	26
4.1.2	Programación por código	28
	4.1.2.1 Apex.....	28
	4.1.2.2 Visualforce Pages	29
	4.1.2.3 SOQL	29
	4.1.2.4 Aura Components	30
	4.1.2.5 APIs de la Plataforma Lightning.....	31
4.1.3	Programación Visual	31

4.1.3.1	Reglas de validación.....	32
4.1.3.2	Process Builder	33
4.1.3.3	Workflows.....	33
4.1.3.4	Flows.....	34
4.1.3.5	Reglas de duplicidad	35
4.2	Conceptos gráficos	36
4.2.1	Formatos de página	36
4.2.2	Gestor de aplicaciones	37
4.2.2.1	Aplicaciones Clásicas	37
4.2.2.2	Aplicaciones Lightning	37
4.2.2.3	Aplicaciones Conectadas.....	38
Capítulo 5 -	Desarrollo del prototipo	39
5.1	Marco de trabajo	39
5.2	Adaptación al gestor de prácticas PSUCM	40
5.2.1	Procedimiento y concepción previa de PSUCM	40
5.2.2	Análisis de requisitos de PSUCM	42
5.2.3	Objetos y campos de PSUCM	54
5.2.4	Aplicaciones de PSUCM	62
5.2.4.1	Aplicación Lightning.....	62
5.2.4.2	Aplicación conectada	65
5.2.5	Automatizaciones de PSUCM	67
5.3	Web-service	69
5.4	Caso práctico del proceso completo	72
Capítulo 6 -	Conclusiones y trabajo futuro.....	73
Chapter -	Introduction	76

Chapter -	Conclusions and future work	80
Apéndice A -	Entrevista a Daniel Chaver	87
Apéndice B -	Manual de configuración OAuth 2.0 JWT Bearer Flow	93

Capítulo 1 - Introducción

En este primer capítulo se presentará la motivación del proyecto, los objetivos planteados al inicio del mismo y el plan de trabajo que se desempeñará para llevar a cabo estos objetivos.

1.1 Motivación

Inmersos en plena pandemia del coronavirus, se ha ensalzado aún más la necesidad de un proceso de digitalización en el que nuestra sociedad lleva mejorando durante las últimas dos décadas. De hecho, según varios CEOs de empresas punteras en diferentes sectores han situado la aceleración de la digitalización entre los 5 y 10 años. Por ejemplo, Nacho de Pinedo, director general de ISDI, ha afirmado lo siguiente: *"Los 60 días de confinamiento han acelerado seis años la digitalización del mundo"* [\[1\]](#). Esto se debe a que uno de los sectores más afectados por la pandemia es el sector empresarial, ya que muchas empresas no están digitalizadas y no han podido vender sus servicios o productos, suponiendo en muchas ocasiones su cierre temporal o definitivo.

Para superar todo lo acontecido, muchos negocios se han apoyado en la digitalización y en el marketing digital. Estos dos conceptos van de la mano en la definición de CRM, una aplicación que permite centralizar en una única base de datos todas las interacciones entre una empresa y sus clientes. El principal CRM del mercado es Salesforce [\[2\]](#), que dispone de una amplísima lista de servicios en la nube que han ido evolucionando y creciendo con el paso de los años.

La unión de la necesidad de digitalización, mi descubrimiento personal con Salesforce y la idea sobre su aplicación en el ámbito académico, me llevó a proponer el presente trabajo para mejorar la plataforma de gestión de las prácticas en la universidad, ya que recientemente tuve que realizar mis prácticas curriculares y trabajé con el GIPE que, sin desmerecer la herramienta, se queda algo escasa y podría aportar una mayor funcionalidad. Este proyecto no es sólo aplicable a la Universidad Complutense de Madrid y se podría vender al resto de universidades mediante una estrategia comercial bajo el nombre de *"Salesforce Practices Project"* [\[3\]](#). Este sería aplicable tanto a

universidades con un gestor obsoleto como universidades que no tienen gestor de prácticas propio, como es el caso de la Universidad de Jaén que actualmente se apoya en la plataforma Ícaro para gestionar la realización de prácticas.

1.2 Objetivos

Para la definición de este proyecto es necesario describir una serie de objetivos que se tratarán de llevar a cabo en el desarrollo del mismo:

- Adaptar el concepto de CRM a una aplicación de gestión de prácticas universitarias.
- Plantear dos interfaces, la primera orientada hacia un usuario con conocimientos de Salesforce (coordinador de prácticas) y la segunda interfaz, ajena a Salesforce, que permita ser usada por alumnos y empresas.
- Usar Salesforce y su API Rest para la creación de la plataforma de prácticas.
- Customizar la plataforma para sus distintos usuarios (coordinador de prácticas, empleados de recursos humanos de cada empresa y alumnos que optan a realizar prácticas).
- Proporcionar al coordinador de prácticas la capacidad de cambiar reglas de validación o añadir valores a la plataforma sin necesidad de programar directamente o pedirlo al equipo de desarrollo.
- Crear una aplicación que establezca la base del proceso de gestión de prácticas, centrándose en la escalabilidad y el crecimiento a futuro que esta pueda tener.
- Diseñar y optimizar los procesos administrativos relacionados con la gestión de prácticas.
- Mejorar la plataforma de gestión de prácticas que la UCM usa actualmente (GIPE).
- Aumentar la seguridad en las transacciones con documentos e información de carácter sensible.
- Estudiar todas las posibles herramientas, que se puedan usar para la mejora de la plataforma, relacionadas con el ámbito de Salesforce. Por ejemplo, Salesforce Blockchain, Hello Sign, Tableau, etc.

1.3 Plan de trabajo

Aquí se describe el plan de trabajo a seguir para la consecución de los objetivos descritos en el apartado anterior, acompañado de un diagrama de Gantt con la planificación prevista.

1. Investigación sobre los distintos CRM (Customer relationship management) del mercado. En esta primera etapa se hizo una búsqueda de los principales CRM, comparándolos con Salesforce, para explicar porque este es considerado el mejor CRM del mercado, además del más usado. En esta fase también se investigan todas las funcionalidades de Salesforce, para seleccionar cuáles son más interesantes para este proyecto.
2. Investigación sobre las distintas APIs y herramientas de integración con Salesforce. Una vez comprobado el poder de mercado que ejerce Salesforce sobre el resto de CRM, en esta segunda etapa se hizo una evaluación de las distintas opciones para el diseño frontend de la aplicación, así como su integración con las distintas APIs de las que dispone Salesforce.
3. Adaptación del modelo de gestión de prácticas a una aplicación con CRM Salesforce. Esta tercera etapa engloba el diseño de objetos, las relaciones entre los mismos y la personalización de la aplicación, con el objetivo de obtener la funcionalidad deseada.
4. Especificación de requisitos. En esta cuarta etapa se definieron todas las funcionalidades de la aplicación.
5. Diseño de la aplicación. En esta quinta etapa se diseñaron los bocetos de la aplicación, tanto de la parte vinculada con Salesforce como la parte de la herramienta front elegida (Angular).
6. Implementación. En esta sexta etapa se procedió al desarrollo de un prototipo funcional de una aplicación programada en Angular y con un backend apoyado en Salesforce. Dicha aplicación realizará la labor de una bolsa de trabajo, usando técnicas y conceptos relacionados con el CRM y con el marketing digital en general.

7. Pruebas y evaluación final. En esta última etapa se realizaron una serie de pruebas para comprobar el correcto funcionamiento de la aplicación, así como hacer un testeo entre un conjunto acotado de alumnos de la UCM para que evaluaran este proyecto con respecto al actual GIPE.

En la *Figura 1-1* se puede apreciar un diagrama de Gantt cuyo objetivo era obtener una imagen global del proyecto ligado a una estimación de las fechas de la realización del mismo.

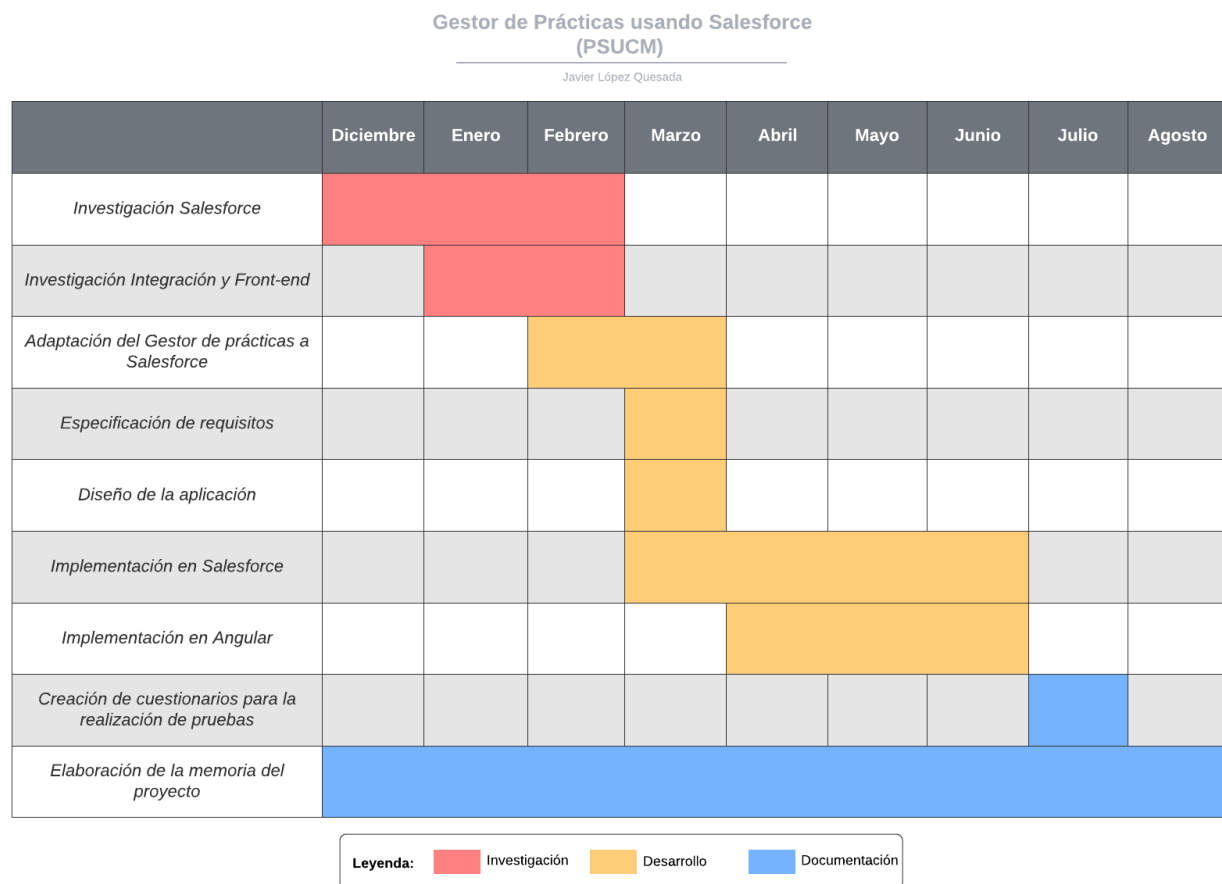


Figura 1-1. Planificación temporal del proyecto 2020/2021

Capítulo 2 - Estado de la cuestión

En este capítulo, se va a presentar el estado del arte realizado para el desarrollo del proyecto. En concreto, se comentarán las técnicas y herramientas que suelen utilizarse para la implementación de sistemas de gestión, tanto para la gestión interna de proyectos como para la gestión de la comunicación. A continuación, se describirán los conceptos de sistemas de gestión de contenidos (CMS), planificación de recursos empresariales (ERP) y gestión de las relaciones con los clientes (CRM) [\[29\]](#), además de ver algunos proyectos que se apoyan en Salesforce, aportando funciones similares a las desarrolladas en este proyecto.

2.1 Sistemas de gestión de contenidos (CMS)

El principal objetivo de estos sistemas es que el administrador de una web sea capaz de añadir, modificar y eliminar textos y contenido multimedia, sin necesidad de contactar continuamente con el programador. Entre los CMS más importantes del mercado destacan Wordpress, Joomla! y Drupal [\[4\]](#). El uso de estos sistemas aporta una gran cantidad de beneficios y ventajas, sin embargo, no son ni mucho menos perfectos. En la siguiente tabla se recogen una serie de ventajas y desventajas de los CMS.

Ventajas	Desventajas
Fácil de usar por personas poco técnicas	Graves problemas de seguridad debido a la falta de actualizaciones
Menos costoso que un desarrollo desde cero	Necesidad de utilizar código para no limitar la funcionalidad, ya que hay algunos procesos que no vienen dentro del estándar.
Implementación rápida de elementos comunes como encuestas, formularios, cuestionarios, mapas y todo tipo de contenido multimedia	Poca flexibilidad en el diseño de la web, ya que necesitas usar código para salir de los diseños más estándar
Reducción en los tiempos de implementación y en el personal de mantenimiento	Necesidad de compra de elementos adicionales (plantillas, plugins, automatizaciones, etc.) para optimizar el sistema y evitar que quede obsoleto

Tabla 2-1. Ventajas y Desventajas de los CMS

Además, dentro de los CMS se puede ubicar el concepto de E-commerce, que se puede definir como una plataforma web enfocada al comercio electrónico. Su uso supondría los mismos pros y contras que cualquier CMS con la diferencia de que estos son mucho más eficientes para plataformas dedicadas a la compraventa de productos. En conclusión, los CMS son muy beneficiosos si se quiere trabajar con sistemas que buscan cubrir las necesidades básicas de una web y que además esta no posea datos de carácter sensible, ya que no tienen una seguridad óptima.

2.2 Planificación de recursos empresariales (ERP)

Esta planificación está orientada a los procesos de backend, entre los que destacan la contabilidad, la facturación, la gestión del stock, la gestión del proyecto y la gestión de recursos. En resumen, todos los procesos internos que se realizan en la empresa y no implican de un proceso comercial con el cliente [\[28\]](#).

Ventajas	Desventajas
Integración en una sola plataforma o aplicación, lo que supone un sistema más seguro y que permite recuperar datos más fácilmente	Para manejar el ERP se necesita formación por parte de todos los trabajadores de la empresa
Automatización y optimización de procesos en la empresa, así como una mejora sustancial en la toma de decisiones	El coste es más elevado que otros sistemas, ya que no es una herramienta que comience desde cero
Se puede dividir en varios módulos independientes, cada uno de ellos con una gran escalabilidad	Se necesita de una conversión de datos sólida y bien definida, ya que si no la hay ralentiza los procesos
Mejores análisis estadísticos en base a las necesidades de cada empresa	Si no hay un trabajo constante a la hora de introducir datos y organizar procesos, la utilidad y eficacia del ERP baja considerablemente

Tabla 2-2. Ventajas y Desventajas de los ERP

Aunque Oracle y SAP son los ERP más utilizados [\[5\]](#), existe uno desarrollado sobre la propia plataforma Salesforce llamado FinancialForce [\[6\]](#). Este se puede integrar de forma muy rápida y sencilla mediante la AppExchange de Salesforce ([3.3.6.1](#)), con el principal inconveniente de que el coste oscila los 500\$ mensuales. En lo que a

funcionalidad respecta, este añade a la plataforma de Salesforce varias herramientas como los sistemas de facturación y contabilidad, la gestión de cobros y recursos y el control de gastos y márgenes.

En conclusión, los ERP son herramientas muy importantes para la gestión interna de una empresa, pero por si solos no constituyen una plataforma sólida para albergar proyectos en los que haya comunicaciones constantes con el cliente. Sin embargo, son un complemento ideal para los CRM y así poder conseguir una visión 360° del cliente.

2.3 Gestión de las relaciones con los clientes (CRM)

CRM (Customer Relationship Management) se define como toda plataforma capaz de soportar procesos comerciales con el cliente. Es decir, todos los procesos van orientados a candidatos, contactos, cuentas, oportunidades y otros objetos relacionados con la gestión comercial. Hay una gran cantidad de CRMs en el mercado [\[7\]](#), muchos de ellos están respaldados por ERPs, aunque esto no va a ser un factor determinante, debido a que este proyecto está más orientado a las comunicaciones entre alumno, empresa y gestor de prácticas, por lo que no se necesita una herramienta interna para generar facturas, cobros, etc.

Como se aprecia en la imagen, Salesforce es la plataforma CRM más importante del mundo. Está muy por encima de sus competidores, tanto en uso como en funcionalidad. Aunque para comprender mejor este gráfico en la *Figura 2-1* se presentan las ventajas y desventajas de los principales CRM del mercado [\[8\]](#).

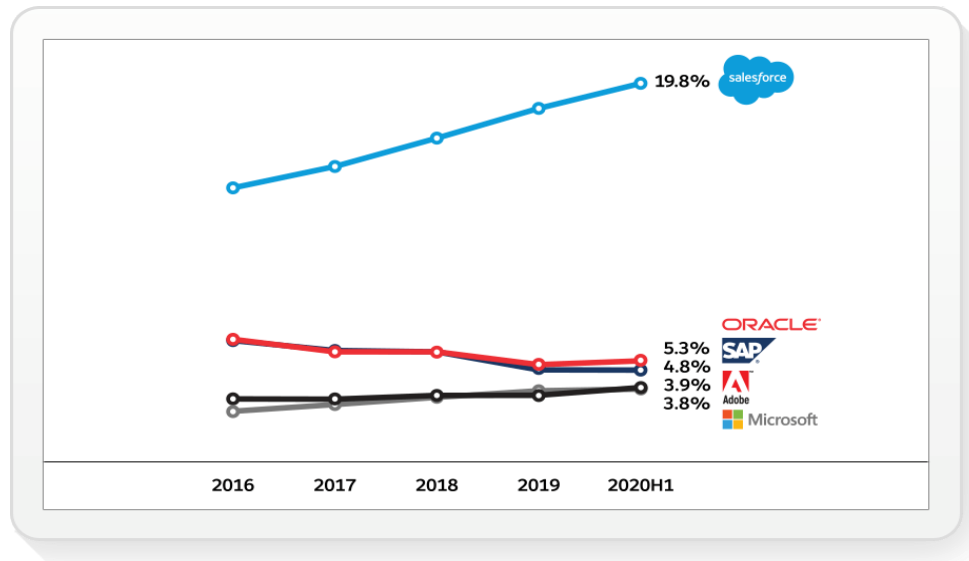


Figura 2-1. Gráfica del porcentaje de uso de los distintos CRMs

2.3.1 Microsoft

Microsoft Dynamics 365 Sales es la plataforma usada por Microsoft para simplificar sus procesos de negocio actuales. Este fue lanzado en 2016 e incluye aplicaciones empresariales inteligentes CRM y ERP. En la siguiente tabla se recoge un resumen de las principales ventajas y desventajas que tiene [\[9\]](#) [\[12\]](#).

Ventajas	Desventajas
A nivel de usuario es sencillo y fácil de usar, incluso para los que no son técnicos	Gran dependencia con los productos de Microsoft, debido a su pobre optimización con otras herramientas más usadas y gratuitas, como por ejemplo las de Google
Puede actuar tanto a nivel local como albergarse de manera íntegra en la nube, manteniendo su uso	Hay que tener muy claras las automatizaciones y como se quiere que funcione la organización, ya que no da muchas facilidades en este aspecto
Se puede dividir en varios módulos independientes, que permiten que se puedan obtener de manera independiente sin necesidad de comprar el paquete completo	Siguen en desarrollo para abarcar tantas funcionalidades como el resto de CRM
Permite código personalizado en plantillas programadas con HTML, Liquid y JavaScript	No permite personalización ni ajustes en el código del servidor

Tabla 2-3. Ventajas y Desventajas de Microsoft Dynamics 365

En conclusión, se trata de un CRM que aún está lejos de convertirse en un referente y que está demasiado encasquillado en los productos que Microsoft ofrece. Además, que su inclusión en el mercado no ha sido tan fuerte debido a que es ahora cuando está apostando fuerte por esta herramienta.

2.3.2 Adobe

Adobe Experience Manager es la plataforma usada por Adobe para simplificar y gestionar sus procesos de negocio actuales. Se sitúa dentro del gestor de contenidos Adobe Experience Cloud, lanzado a finales de 2012. En la siguiente tabla se recoge un resumen de las principales ventajas y desventajas que tiene [\[10\]](#) [\[13\]](#).

Ventajas	Desventajas
Es capaz de almacenar una gran cantidad de contenido	A nivel de usuario es difícil de usar si no se tiene un perfil técnico
Muy enfocado a la mejora en la gestión de tareas y la inteligencia artificial	El coste es muy elevado para la mayoría de las empresas pequeñas
Proporciona una gestión de elementos multimedia, entre los que destaca la inclusión del formato audiovisual.	Se necesitan varios inicios de sesión para acceder a las distintas áreas del sistema, lo que supone una pérdida en la productividad de empleados que accedan a varias áreas
Permite usar la misma solución en páginas webs distintas	Poca personalización y cierta rigidez en la mayoría de las automatizaciones

Tabla 2-4. Ventajas y Desventajas de Adobe Experience Manager

En conclusión, si se utilizan productos Adobe y se quiere tener un CRM más parecido a un CMS, capaz de trabajar con contenido multimedia, es sin duda la mejor opción. Por el contrario la rigidez y el coste son impedimentos para decantarnos por él en este proyecto.

2.3.3 SAP

SAP CRM es una plataforma usada por SAP SE para simplificar y optimizar procesos de negocio. Es uno de los más antiguos ya que se comenzó a desarrollar en el año 2007. En la siguiente tabla se recoge un resumen de las principales ventajas y desventajas que tiene. En la siguiente tabla se recoge un resumen de las principales ventajas y desventajas que tiene [\[5\]](#) [\[14\]](#).

Ventajas	Desventajas
Buen nivel de personalización, permitiendo una buena adaptación con las necesidades de la empresa	Envejecimiento general debido a que su interfaz y funcionalidad se ven obsoletas. Además, no incluye tantos módulos con nuevas tecnologías
Optimización en la generación de informes, reduciendo el tiempo de creación y aportando inteligencia de negocio	La sobrecarga de opciones en la interfaz repercute en la facilidad de uso, haciendo que se requiere de una buena formación
Gran capacidad de integración, de forma nativa, con la gran mayoría de aplicaciones existentes	Las búsquedas son algo lentas y pesadas, especialmente a nivel de base de datos
Se puede dividir en varios módulos independientes, que permiten que se puedan obtener de manera independiente sin necesidad de comprar el paquete completo	La integración con el backend no es muy fluida, lo que requiere de un esfuerzo añadido

Tabla 2-5. Ventajas y Desventajas de SAP CRM

En conclusión, se trata de un CRM que siempre ha estado ahí pero que se está quedando algo desfasado con la continua actualización de Salesforce y la aparición de Microsoft y Adobe en el panorama CRM.

2.3.4 Oracle

Oracle Sales Cloud es la plataforma usada por Oracle para simplificar sus procesos de negocio actuales. Dentro de sus distintas aplicaciones la encargada de la parte CRM es Oracle Cloud CX, aunque también tiene una aplicación dirigida a ERP. Fue lanzado en 2016, aunque en 2018 presentaron su segunda generación de aplicaciones. En la

siguiente tabla se recoge un resumen de las principales ventajas y desventajas que tiene [\[5\]](#) [\[15\]](#).

Ventajas	Desventajas
Tiene muy buena organización en los datos de clientes, así como en las relaciones entre elementos	La interfaz de usuario está obsoleta y necesita una apariencia más moderna
Uso de la inteligencia artificial para mostrar información relevante al usuario	El aprendizaje de su inteligencia artificial es limitado y no permite modificaciones
Permite la integración con Salesforce mejorando la interrupción de negocios con este CRM	Los informes no son completos y presentan poca automatización
La asistencia interna es muy buena	Tiempo de instalación, costos y mantenimiento significativos.

Tabla 2-6. Ventajas y Desventajas de Oracle Cloud CX

En conclusión, es un CRM muy válido y eficiente. Sin embargo, aún está muy por debajo de Salesforce en funcionalidad y la integración con el mismo hace que sea un buen complemento de este en lugar de un sistema independiente.

2.3.5 Salesforce

Salesforce CRM es la plataforma usada por Salesforce.com para simplificar, optimizar y automatizar procesos de negocio. Su proyecto CRM comenzó en 2004, incluyendo año tras año nuevas herramientas y nubes que lo hicieron el más popular y usado del mundo. En la siguiente tabla se recoge un resumen de las principales ventajas y desventajas que tiene [\[11\]](#) [\[16\]](#).

Ventajas	Desventajas
A nivel de usuario es sencillo y fácil de usar, incluso para los que no son técnicos	Coste bastante elevado y limitado a nivel de usuario
Es un CRM completo englobando todas las acciones con cliente, permitiendo además un nivel de personalización muy alto	Limitaciones en correos, registros y otras funcionalidades, según la versión comprada

El inicio a nivel de formación y trabajo es mucho más rápido que en el resto de CRMs, permitiendo diseñar y gestionar un sistema funcional en poco tiempo	Se utiliza el mismo mainframe para todos los usuarios, considerándose un sistema de inquilino único. Esto hace que puedan existir colisiones cuando dos usuarios realizan o modifican una tarea similar
Es un sistema muy seguro que cuenta con métodos muy innovadores en la protección de datos, permitiendo ampliar el nivel de seguridad con módulos extra	Algunas funcionalidades o aplicaciones no se pueden probar o testear hasta que hayan sido compradas, lo que supone que no se sepa de la eficacia real de estas hasta haber adquirido el producto

Tabla 2-7. Ventajas y Desventajas de Salesforce

En conclusión, Salesforce es la mejor opción en el mundo de los CRM siempre y cuando exista una disponibilidad económica, debido a que sus principales problemas se solucionan comprando una cuenta Unlimited que supone un mayor desembolso. Por lo demás, es muy completa y, al contrario que muchos de los CRM citados anteriormente, proporciona una cuenta de developer totalmente gratuita, para practicar o testear ciertos proyectos [\[17\]](#).

Fijándonos en datos de la web *CompareCamp* [\[12\]](#) [\[13\]](#) [\[14\]](#) [\[15\]](#) [\[16\]](#), uno de los principales recursos en línea de reseñas y comparaciones de productos, las puntuaciones que nos proporciona sobre los distintos CRMs son las expuestas en la siguiente tabla.

	Salesforce	Oracle	SAP	Adobe	Microsoft
Precio inicial	5\$	65\$	Según presupuesto	Según presupuesto	40\$
Versión gratuita	✓	✗	✓	✗	✗
Características	9.4	8.8	9.3	9.6	9.4
Facilidad de uso	8.9	8.9	9.0	9.4	9.2
Asistencia al cliente	9.0	8.9	8.9	9.4	8.9
Valor del dinero	9.2	9.0	8.8	9.5	9.3
Porcentaje de uso	19.8%	5.3%	4.8%	3.9%	3.8%

Tabla 2-8. Comparativa general entre los distintos CRM

Aparentemente los CRM de Microsoft y Adobe tienen mejores puntuaciones que el propio Salesforce. Sin embargo, hay algunos factores que hacen que Salesforce sea una mejor opción tanto para las empresas como para este proyecto.

- 1) La variedad de versiones de Salesforce le permite un precio de partida muy por debajo de sus competidores.
- 2) Tiene una versión gratuita ideal para este tipo de proyectos y las limitaciones de la misma no afectan a la mayoría de las funcionalidades.
- 3) Salesforce sigue siendo el CRM más usado con casi un 20% del mercado, lo que hace que haya más información en la web para una mejor formación.
- 4) Dynamics 365 o Adobe Experience Manager están valoradas suponiendo que la empresa trabaja principalmente con productos Microsoft o Adobe, lo que optimiza mucho el CRM correspondiente pero no siempre se da.
- 5) Las puntuaciones en las que se ve superado no suponen más de 0,5 puntos por encima, por lo que no hay una diferencia notoria y se suelen centrar en la facilidad de uso, lo cual con una buena formación se solventa fácilmente.

2.4 Proyectos y artículos relacionados con Salesforce

A continuación, se verán una serie de artículos y proyectos, con contenido y bibliografía científico-académica, sobre los CRM y Salesforce. Estos son de vital importancia para el desarrollo del proyecto ya que exponen las ventajas sobre el uso de estas tecnologías en ámbitos como el académico o a nivel de integraciones. Estos son:

- *“Feature Analysis of the “CustomerRelationship Management” Systems for Higher Education Institutions”* [\[20\]](#). Este artículo relaciona el ámbito educativo con el mundo de la gestión de las relaciones con los clientes, proponiendo CRM como la solución ante la alta demanda por parte de padres, alumnos, gestores y profesores. Estos exigen más atención a través de diferentes canales de comunicación, con una respuesta casi inmediata, y un servicio con intermediarios pertenecientes a las distintas instituciones implicadas. El objetivo del artículo es enumerar una serie de características clave, que debe poseer un CRM, para que

las instituciones educativas puedan mejorar y prosperar durante su transformación digital.

- “*Salesforce Integration in New Product Development – A Key Driver of New Product Success?*” [\[21\]](#). En este artículo se demuestra a través del análisis de datos y resultados la acogida de Salesforce, considerándolo un impulsor clave en el éxito de nuevos productos comerciales. Para ello se analizan tres factores que son; la calidad de la información, el tiempo que tarda en proporcionar u obtener la información y los posibles errores que puede generar esta plataforma. La conclusión es muy positiva, destacando el gran rendimiento de Salesforce como CRM.
- “*The Next Wave of CRM Innovation: Implications for Research, Teaching, and Practice*” [\[22\]](#). En este artículo se analizan las principales innovaciones que se están produciendo en el plano orientado a la gestión de relaciones con el cliente, haciendo hincapié en sectores como el Big Data, Internet de las Cosas, Analytics e Inteligencia Artificial. Además, incluye algunas ideas sobre investigaciones futuras y posibles innovaciones que son muy interesantes de estudiar y analizar.
- “*Implementación de un CRM con tecnología Salesforce aplicado a la gestión de prácticas externas de la Escuela Politécnica de Cuenca*” [\[23\]](#). Trabajo fin de grado que trata sobre la adaptación de Salesforce a la gestión de prácticas externas, pero sin realizar ninguna integración web. Aprovecha los elementos estándar para simular cómo sería un proceso de gestión de prácticas usando este CRM, aunque evita una gran personalización y desarrollo de automatizaciones del propio sistema.
- “*Application of SalesForce Platform in Online Teaching in Colleges and Universities under Epidemic Situation*” [\[24\]](#). Este estudio es de los pocos realizados durante la pandemia y se centra en el análisis de los servicios que proporciona Salesforce para la gestión del trabajo en línea. Se centran en el ámbito de la enseñanza (colegios y universidades) para afrontar cualquier pandemia sin que suponga un condicionante en la enseñanza de los alumnos.

Capítulo 3 - Tecnologías empleadas

En este capítulo se describen todas las tecnologías empleadas para la creación de este proyecto, las cuales se pueden agrupar en: tecnologías utilizadas para la interfaz del usuario (front-end), las usadas para la lógica del proyecto (back-end) y otras herramientas que ayudan al desarrollo del mismo.

3.1 Tecnologías usadas para frontend

Entendemos por “*Front End*” la parte de una aplicación que interactúa con los usuarios. En este proyecto no hay un único usuario hacia el que va dirigido la aplicación, por lo que también hay dos herramientas de front. Por un lado, la parte web, usada por empresas y alumnos, usa Angular, mientras que la parte de Salesforce es usada por el coordinador de las prácticas.

3.1.1 Angular

Angular es un marco de diseño de aplicaciones y una plataforma de desarrollo para crear aplicaciones. Se sitúa como una de las principales herramientas front-end en el mundo del desarrollo software, además en él se usan varios lenguajes de programación punteros, así como bibliotecas para facilitar la implementación [\[25\]](#). En el apartado gráfico, es muy común encontrar plantillas gratuitas sin funcionalidad que permiten tener una base de la que partir, como es el caso de este proyecto [\[27\]](#).

3.1.1.1 HTML5

Se trata de la última versión de HTML que aporta un conjunto más amplio de tecnologías. Además, posee nuevos elementos, atributos y comportamientos que facilitan la tarea a los desarrolladores con respecto a su antecesor.

3.1.1.2 TypeScript

TypeScript ofrece todas las funciones de JavaScript con la ventaja de que, a diferencia de su antecesor, este posee un sistema de tipos. De esta forma TypeScript comprueba

que las asignaciones de variables se realizan de manera coherente, mejorando la programación y permitiendo detectar los errores con mayor facilidad.

3.1.1.3 SASS

Sass es un lenguaje de hoja de estilos que se define como “CSS with superpowers” por lo que puede ser considerado el sucesor del ya conocido CSS. Con este lenguaje la aplicación consigue un aspecto visual moderno que la hace más atractiva de cara al público más innovador.

3.1.1.4 Bootstrap

Es una biblioteca multiplataforma de código abierto ideal para mejorar la parte frontend de una aplicación web. Bootstrap incluye variables, componentes prediseñados y potentes complementos de JavaScript, por lo que es una biblioteca perfecta para la creación de formularios, diseño de la aplicación y optimización de código.

3.1.2 Salesforce

Salesforce se diseñó como una herramienta con un acabado frontend para que fuese usada de manera rápida y sencilla por personas con conocimientos de marketing. Es por eso que incluye una gran cantidad de herramientas y funcionalidades para mejorar la visibilidad de objetos, campos, botones, acciones, etc. Las herramientas más asociadas al aspecto “front” de Salesforce se explicarán con más detalle en el punto [4.2](#).

3.2 Tecnologías usadas para backend

Entendemos por “Back End” el interior de una aplicación, normalmente formado por un servidor, una aplicación y una base de datos. En un primer momento se barajó la posibilidad de incluir un backend propio con NodeJS apoyándose en una biblioteca, para su integración con Salesforce, llamada JSforce. Aunque esta idea se descartó debido a que el tiempo de creación de un backend bien diseñado se aleja del alcance real del proyecto y estaría más enfocado a un proyecto con un equipo de desarrollo completo. Finalmente, en este proyecto no hay una herramienta de backend pura, sino

que sería el propio Salesforce el que se encargaría de la lógica, además de actuar como una base de datos de los registros, archivos y demás contenido multimedia. Todos los aspectos relacionados con la organización y gestión de los datos de Salesforce se explicarán con más detalle en el punto [4.1](#).

3.3 Otras herramientas

Además de las tecnologías ya mencionadas hay muchas otras herramientas de apoyo para mejorar la implementación. Estas van desde entornos de programación hasta herramientas gráficas, la combinación de todas ellas son las que hacen que la aplicación tenga un buen acabado.

3.3.1 Salesforce Developer Edition

Aunque no es una herramienta como tal, en toda la estructura de Salesforce se va a tener acceso con una cuenta especial para desarrolladores. Las cuentas de tipo developer edition proporcionan acceso a la plataforma Lightning y las API, permitiendo ampliar Salesforce integrándose con otras aplicaciones. Además, proporciona acceso a muchas de las funciones disponibles en Enterprise Edition (versión para empresas).

El motivo principal de esta elección es que Salesforce no es una herramienta gratuita como lo son el resto de herramientas utilizadas y por tanto usar una cuenta *Essentials*, *Professional*, *Enterprise* o *Unlimited* durante los meses de desarrollo supondría un desembolso de 25, 75, 150 o 300€ al mes en función de la cuenta elegida [\[17\]](#).

Algunas de las limitaciones a tener en cuenta de la developer edition son:

- Límite de 2 usuarios.
- Almacenamiento de datos de 5 MB.
- Almacenamiento de archivos (adjuntos) de 20 MB.
- Límite de 5000 llamadas API cada 24 horas.
- Uso con fines no comerciales.

La única limitación que afectaría al correcto desarrollo del prototipo sería el límite de usuarios, ya que esto impediría que cada alumno o empresa tuviese acceso a su propia sesión de Salesforce. Sin embargo, se podrían usar cada uno de los registros para

almacenar la clave encriptada de cada alumno o empresa y que con ello pudieran hacer llamadas al API a través de la aplicación front-end.

3.3.2 Github

Github es un servicio basado en la nube que aloja Git como sistema de control de versiones. Se trata de una herramienta clave para cualquier desarrollo, ya que permite recuperar código, haciendo que la implementación sea más segura. Además, en caso de avería en el equipo siempre habrá una copia de seguridad en Github para que no se pierda nuestro progreso. El repositorio del proyecto se encuentra en [\[26\]](#).

Cabe comentar que en este proyecto también se ha usado Github Desktop, la aplicación de escritorio de Github, que facilita aún más la gestión del código.

3.3.3 Visual Studio Code

Es un entorno de desarrollo integrado para crear y depurar aplicaciones web y aplicaciones en la nube. Es compatible con Git (tiene una consola integrada) y posee una gran cantidad de plugins para facilitar aún más la programación. Concretamente para este proyecto el paquete más útil es el llamado "Salesforce Extension Pack" [\[31\]](#) que incluye una gran cantidad de funcionalidades para implementar código proveniente de Salesforce. Con este plugin podremos traernos, a nuestro repositorio, proyectos enteros realizados en nuestra entidad de Salesforce, ya que, aunque posee un entorno de desarrollo propio, este se queda algo corto y no tiene tantas funcionalidades como Visual Studio Code. El proyecto de Salesforce se encuentra en [\[33\]](#).

3.3.4 Postman

Se trata de una aplicación de escritorio muy útil para la gestión de APIs, entre sus funciones están:

- Creación de peticiones a APIs internas o de terceros.
- Elaboración de tests para validar el comportamiento de APIs.
- Posibilidad de simular diferentes entornos de trabajo.

En este proyecto Postman se ha usado para hacer llamadas a las diferentes APIs y realizar múltiples pruebas con distintas casuísticas. Al no requerir de toda una infraestructura, es mucho más fácil que si se tuviera que recrear en el código fuente, ya que se ahorra la creación de variables, mostrarlo por pantalla, acceder a la página en concreto, etc. Ejemplo de llamada en Postman en la *Figura 3-1*.

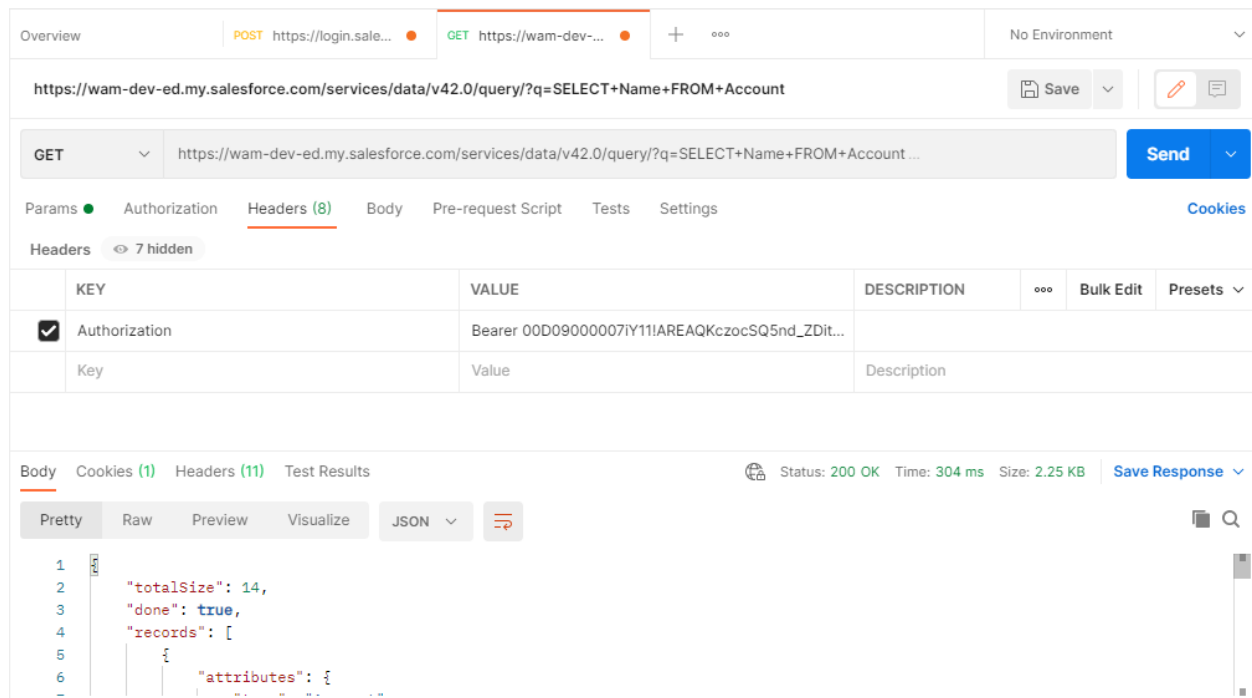


Figura 3-1. Ejemplo de llamada GET en Postman

3.3.5 Beeceptor

Se trata de una herramienta web online con multitud de funcionalidades entre las que destaca inspeccionar el tráfico HTTP. En este proyecto es muy útil comparar las peticiones que manda Angular y las que manda Postman, ya que Beeceptor incluye soporte CORS para evitar el bloqueo de peticiones por parte de este protocolo.

Para usar esta aplicación correctamente hay que definir un nombre para la creación de un endpoint, de manera que las peticiones post, get o put se realicen a dicho endpoint, capturando esta la información y mostrando lo que realmente estamos enviando. Ejemplo de llamada en Beeceptor en la *Figura 3-2*.

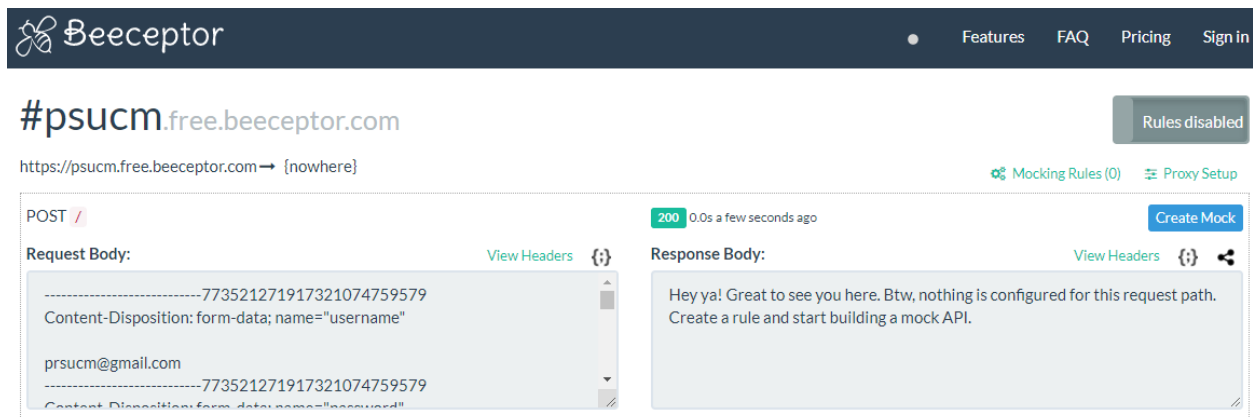


Figura 3-2. Ejemplo de llamada POST recibida en Beeceptor

3.3.6 Complementos para Salesforce

Hay muchas herramientas relacionadas con el mundo Salesforce que ayudan a mejorar la experiencia de este CRM. Entre las cientos de funcionalidades, a continuación se describirán las usadas en este proyecto.

3.3.6.1 Salesforce AppExchange

Dentro del ámbito de Salesforce existe un Marketplace [\[32\]](#) de aplicaciones y paquetes para mejorar y ampliar algunas funcionalidades de Salesforce. Para este proyecto se han usado los siguientes paquetes gratuitos:

- **CMTD Enhanced Related List:** Su principal funcionalidad es la de proporcionar listas relacionadas mejoradas basadas en metadatos, de manera que permite personalizarlas proporcionando una interfaz más intuitiva.
- **Simple Signature:** Se trata de un paquete gestionado que permite capturar firmas electrónicas para usarlas con facilidad a la hora de generar documentos. Además incluye páginas de visualforce, procesos y elementos que mejoran la seguridad de los documentos.

3.3.6.2 Salesforce Inspector

Salesforce Inspector es una extensión de Google Chrome muy útil para el desarrollo de cualquier aplicación, proceso o diseño dentro de Salesforce. Principalmente permite

realizar consultas SOQL de manera sencilla, importar datos de forma similar a Data Loader y revisar el contenido de ciertos campos a los que no se tiene acceso mediante el layout. Todo esto hace que Salesforce Inspector sea vital para agilizar y mejorar el desarrollo en Salesforce.

3.3.6.3 Data Loader

En Salesforce es muy habitual que se introduzcan registros de forma masiva, ya que rellenar cada campo de manera individual puede ser una tarea muy tediosa. Sin embargo, existe una herramienta llamada Data Loader cuya función es insertar, actualizar o eliminar registros de forma rápida y segura. Para ello, realiza un mapeo entre un conjunto de datos, previamente almacenados en formato csv, y los campos asociados al objeto de Salesforce, de forma que dada una clave (normalmente el id) asigna, actualiza o elimina campos y registros.

En este proyecto el Data Loader está ligeramente capado, ya que al tratarse de una cuenta gratuita (Developer Account) no se puede almacenar más de 5 MB en datos, no obstante no influye en el desarrollo del prototipo ya que con un bajo número de registros se pueden realizar las pruebas correspondientes.

3.3.6.4 Funciones Einstein

Las funciones Einstein proporcionan a Salesforce una gran cantidad de automatizaciones, uso de inteligencia artificial y herramientas de seguridad. Existen muchísimas funciones pero las usadas en este proyecto son el Einstein Activity Capture cuya función es vincular y mantener actualizados los datos entre una cuenta vinculada (Google, Office365 o MS Exchange) y Salesforce. Se puede filtrar a nivel de usuario y en base a los registros de correos electrónicos, calendario y contactos. De esta forma en la aplicación las reuniones definidas en el calendario de Google crearán tareas en Salesforce indicando estos eventos..

3.3.7 Wix

Wix es una plataforma basada en la creación de páginas webs que utiliza como lenguaje de programación HTML5. No hay que confundirlo con un CMS, ya que los

creadores de sitios web proporcionan alojamiento propio, mientras que en los CMS el alojamiento debe ser obtenido por tu cuenta. Este es el principal motivo para usar Wix por encima de otros sistemas de gestión de contenidos definidos en el punto [2.1](#).

La popularidad de Wix viene dada por la facilidad de uso y creación que ofrece al usuario. Wix se adapta al creador de la web ofreciendo la opción de crear a través de código HTML5 o con el editor web que integra la plataforma basado en arrastrar y soltar, de esta manera cualquier persona puede desarrollar su web tenga o no conocimientos de programación. En este proyecto se ha usado para la creación de una landing page (página de aterrizaje) cuya finalidad es informar sobre el producto que se quiere vender y a cambio, obtener datos del visitante o cliente potencial a través de un obsequio.

3.3.8 Editores gráficos

El grafismo de este proyecto se apoya en varias herramientas gratuitas y de pago que se pueden resumir en:

- **Adobe Photoshop:** La mayoría de los diseños gráficos tanto en la aplicación como en los diagramas incluidos en esta memoria están realizados usando Adobe Photoshop, el software más popular para la edición de fotografías.
- **Lucidchart:** Se trata de una de las herramientas de diagramación basada en web más populares. En el proyecto se ha usado para la creación de los diagramas más ligados a la ingeniería del software (casos de uso o UML).

Capítulo 4 - Salesforce CRM

Como hemos visto en el punto [2.3](#), Salesforce es la plataforma CRM (Customer Relationship Management) más importante del mundo. Está muy por encima de sus competidores, tanto en uso como en funcionalidad y además proporciona una versión gratuita ideal para los futuros desarrolladores. Sin embargo, Salesforce no es una plataforma ni mucho menos sencilla debido a su enorme cantidad de funcionalidades, ya que lleva año tras año añadiendo nuevas herramientas y usos a su sistema. En muchas ocasiones, estas actualizaciones suponen la inclusión de nuevas nubes, que ya se dividen en Sales, Service, Marketing, Analytics, Integration, Commerce, Community, Financial Services, Higher Education, etc. Sin embargo, este proyecto se centrará en la Salesforce Sales Cloud debido a que es la nube más consolidada y se centra en la administración con clientes. En este capítulo se resumen brevemente los conceptos y características más relevantes, de Salesforce, de cara a la realización de este proyecto. Muchas de las referencias se pueden localizar en la documentación oficial de Salesforce [\[17\]](#), [\[18\]](#) y [\[19\]](#), aunque no toda esta información es totalmente pública, ya que se debe tener una cuenta de desarrollador (gratuita) para acceder.

4.1 Conceptos básicos

Antes de comenzar con la adaptación de este CRM al gestor de prácticas, hay que ver algunos conceptos básicos claves para el desarrollo del mismo. Para ello se dividirá este bloque en tres partes, el generador de aplicaciones Lightning que está estrictamente ligado al desarrollo de layouts, campos y en general elementos visuales, la programación por código que añade las partes del desarrollo más personalizables y la parte de programación visual que nos permite programar de forma rápida y sencilla algunas tareas más estandarizadas pero igualmente claves en la gestión de cualquier aplicación.

4.1.1 Organización de elementos

Salesforce tiene una estructura muy sólida que permite albergar una gran cantidad de usuarios que la controlan, otorgando siempre mucha seguridad y una gran funcionalidad. En este apartado se analizará esta estructura y la forma en la que se administra este CRM.

4.1.1.1 Estructura

Pese a ser un CRM de uso comercial, la estructura de Salesforce no difiere mucho de la de cualquier sistema basado en tablas o programación orientada a objetos. Para entender su organización hay que tener muy en cuenta una serie de conceptos sobre los que se realizan todas las acciones [\[17\]](#), [\[18\]](#) y [\[19\]](#). Estos son:

- **Objeto:** Tipo de dato que desea almacenar. Existen objetos estándar (Candidato, Cuenta o Oportunidad) y objetos personalizados que uno mismo puede diseñar y crear.
- **Campo:** Parte de datos almacenados en un objeto. Existen campos estándar (Compañía, Nombre de la cuenta o Etapa) y campos personalizados que uno mismo puede diseñar y crear.
- **Registro:** Cada una de las instancias de un objeto.
- **Relación:** Los registros se relacionan entre sí con la creación de estos campos, permitiendo que un objeto esté relacionado con uno o más registros de otro objeto distinto. Esto permite la creación de las llamadas listas relacionadas. Existen tres tipos de relaciones:
 - **Relación principal-detalle:** Relaciona un objeto subordinado y un objeto principal, siendo este último el que determina la obligatoriedad, propiedad, eliminación y colaboración de los registros subordinados.
 - **Relación de búsqueda:** Vincula un objeto a otro objeto mediante un campo búsqueda donde se selecciona un registro de la lista.
 - **Relación de búsqueda externa:** Vincula un objeto a otro objeto externo cuyos datos están situados fuera de Salesforce.

- **Herencia:** Un registro puede convertirse en el padre de otro mediante la creación de un campo Hierarchy (jerarquía) en el objeto.
- **Objeto cruzado:** La relación muchos a muchos (many to many) es muy recurrente en la creación de proyectos, sin embargo, las relaciones citadas anteriormente no permiten esta relación. Para solventar esta necesidad, Salesforce propone los llamados objetos cruzados (junctions objects), objetos personalizados compuestos por dos campos de tipo relación, cada uno apunta a los objetos que se quieren vincular entre sí.
- **Tipo de metadato personalizado:** Se tratan de datos que describen otros datos, siendo un tipo de dato que alberga valores para una serie de campos creados.

Para alguien con conocimientos en programación orientada a objetos, estos conceptos son muy fáciles de asimilar, ya que, a excepción de algunos términos como campo o registro que se suelen sustituir en POO por atributo o instancia, la terminología y relaciones son idénticas.

Además, Salesforce ya viene con una organización jerárquica de objetos y campos estándar (*Figura 4-1.*). Estos se relacionan entre sí y tienen algunos procesos ya automatizados. Entre estos objetos se encuentran contactos, candidatos, cuentas, oportunidades, tareas, campañas, informes, contratos y un largo etcétera. Esto es beneficioso como base para cualquier sistema, aunque a menudo presenta problemas ya que hay campos estándar que no puedes eliminar o automatizaciones que actúan como una caja negra, siendo incontrolables.

En resumen, un usuario comercial trabajará con una vista (Figura 4-2.) en la que tendrá acceso a los diferentes registros y podrá realizar todas las automatizaciones creadas para esa aplicación.

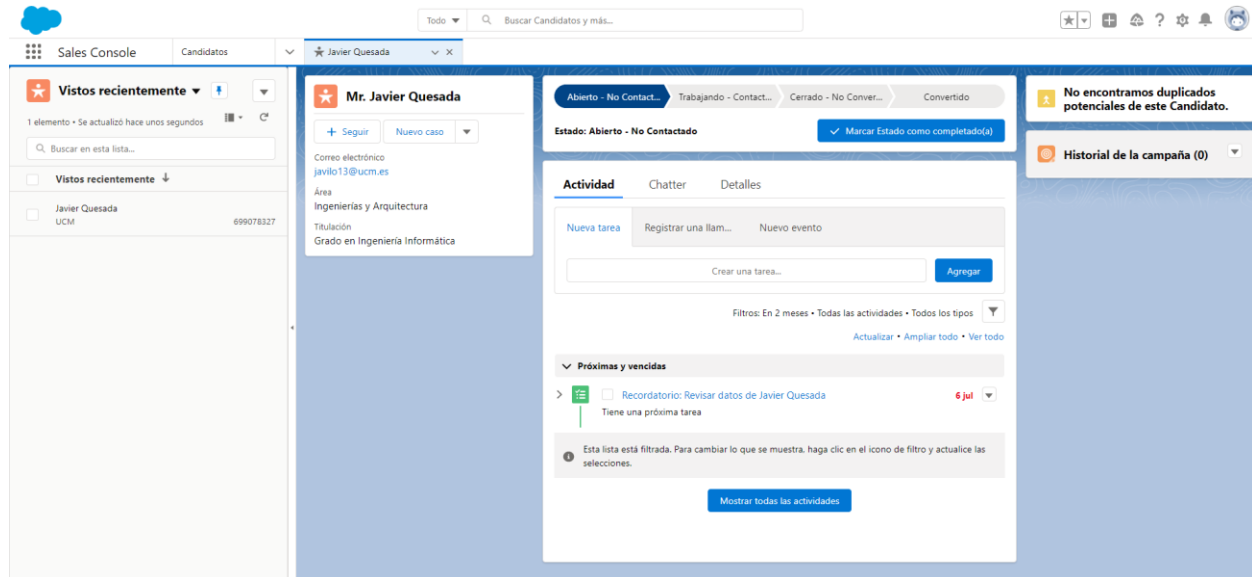


Figura 4-2. Ejemplo Vista Comercial

Un administrador del sistema también tendrá acceso a la vista anterior para realizar pruebas, pero la vista principal con la que trabajará será la que está asociada a la configuración. Con ella tendrá acceso a todos los elementos que forman el sistema como el gestor de objetos, el gestor de aplicaciones, la automatización de procesos, la consola de desarrollador, etc.

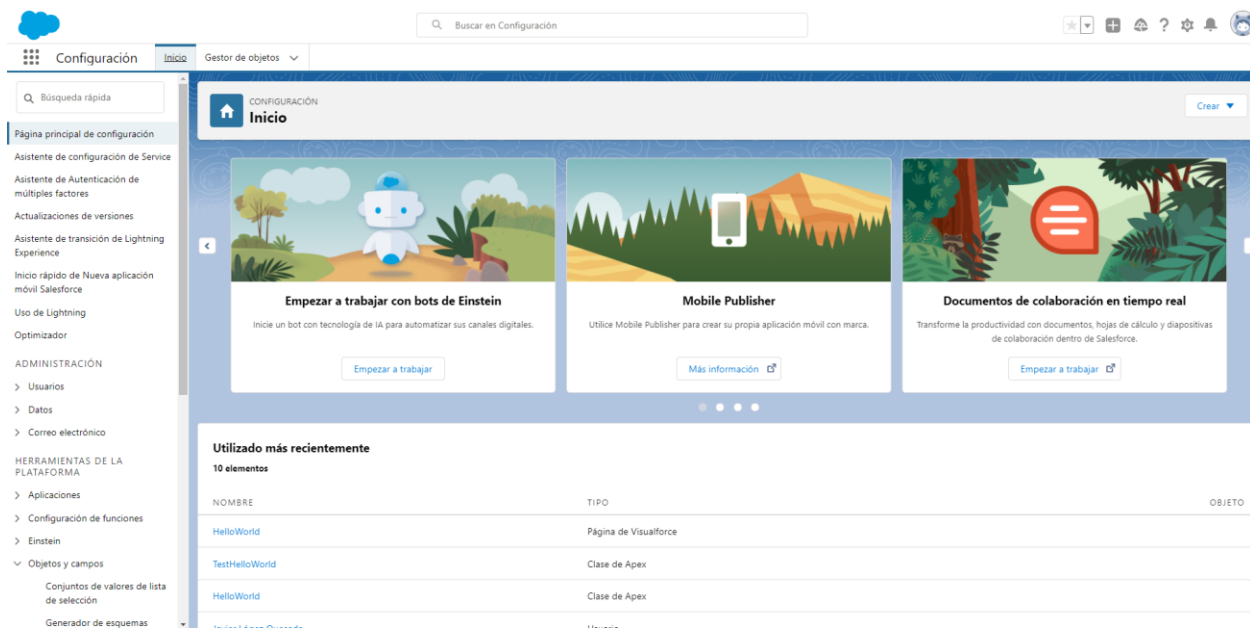


Figura 4-3. Ejemplo Vista Administrador

Mediante permisos se puede limitar el uso de funcionalidades o acceso a la configuración para ciertos roles o perfiles, sin embargo, debido a la limitación de usuarios que tiene la cuenta developer sólo se trabajará con el usuario administrador del sistema.

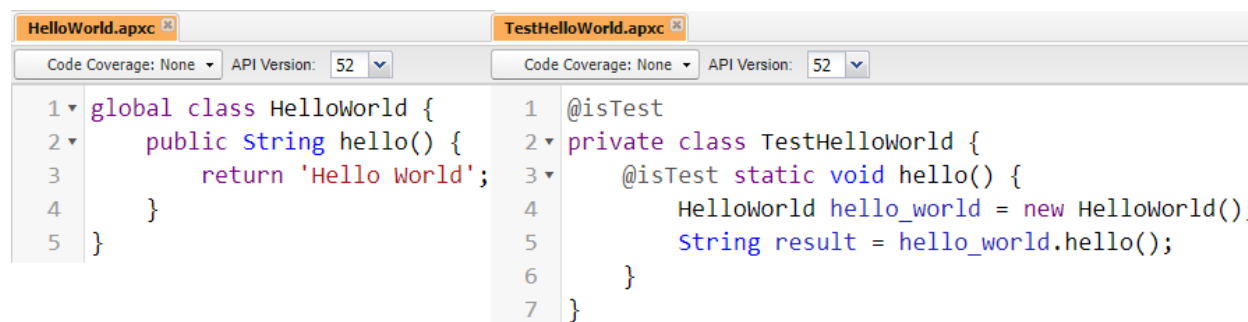
4.1.2 Programación por código

Uno de los fuertes de Salesforce como CRM es que tiene lenguajes exclusivos de la plataforma que se asemejan mucho a otros que ya conocemos como son Java, HTML o SQL. Además, Salesforce cuenta con su propia consola de desarrollador donde se pueden implementar clases, pasar tests o revisar errores de compilación y ejecución [18].

4.1.2.1 Apex

Se trata de un lenguaje de programación orientado a objetos exclusivo de la plataforma Salesforce, cuya sintaxis es muy similar a Java. Este código suele ser usado en clases apex y lanzado a través de solicitudes web o mediante triggers. Además, Apex permite crear clases test y agregar lógica a otros elementos de Salesforce como registros,

botones, recursos estáticos o páginas de Visualforce. Estas clases pueden ser creadas en la consola de desarrollador y tienen extensión .apxc.



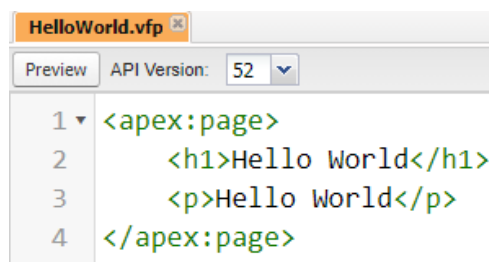
```
1 global class HelloWorld {
2     public String hello() {
3         return 'Hello World';
4     }
5 }

1 @isTest
2 private class TestHelloWorld {
3     @isTest static void hello() {
4         HelloWorld hello_world = new HelloWorld();
5         String result = hello_world.hello();
6     }
7 }
```

Figura 4-4. Ejemplo Hello World Apex

4.1.2.2 Visualforce Pages

Las páginas de Visualforce usan un lenguaje muy similar a HTML, pero incluyendo potentes funciones ETL (extracción, transformación y carga) sobre los datos de una organización. Permiten un acabado muy similar al de muchas webs pero dentro del propio Salesforce, pudiendo usarlas para una mejor visualización de la interfaz estándar de la plataforma. Estas páginas pueden ser creadas en la consola de desarrollador y tienen extensión .vfp.



```
1 <apex:page>
2     <h1>Hello World</h1>
3     <p>Hello World</p>
4 </apex:page>
```

Figura 4-5. Ejemplo Hello World Visualforce

También son muy útiles para crear plantillas y mostrar la propia página usando los “Sites” de Salesforce que le asignan su propio dominio, permitiendo modificar su accesibilidad y permisos.

4.1.2.3 SOQL

Es el principal lenguaje de consulta de objetos en Salesforce y su sintaxis es muy similar a SQL. Las palabras reservadas son las mismas en ambos lenguajes y se diferencian en

cosas puntuales, por ejemplo, SOQL no permite consultar todos los campos de un objeto usando `SELECT *`, consulta muy recurrente en SQL. A pesar de estas diferencias, a efectos prácticos son muy similares y presentan las mismas funcionalidades. Cabe comentar que una persona menos técnica puede no usar SOQL y sustituirlo por la herramienta encargada de generar informes, sin embargo este lenguaje es mucho más rápido.

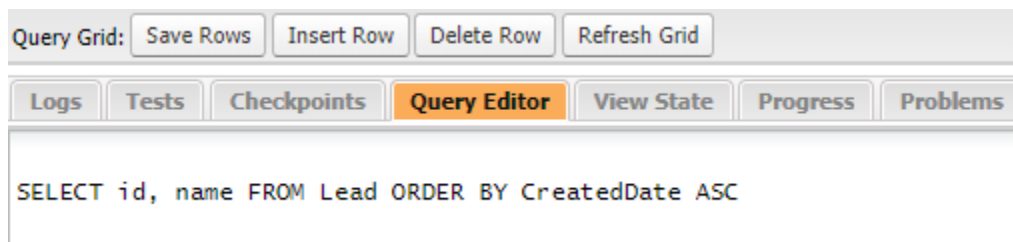


Figura 4-6. Ejemplo SOQL

4.1.2.4 Aura Components

Salesforce tiene un marco de componentes Lightning encargado del desarrollo de aplicaciones web dinámicas de página única, cuya estructura es la siguiente.

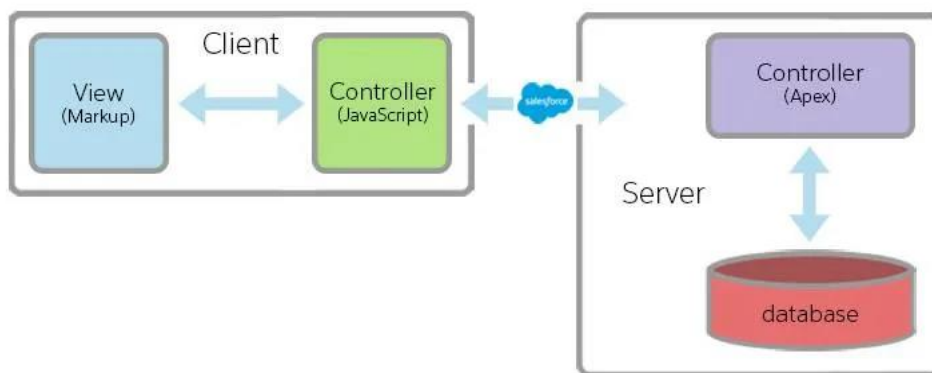


Figura 4-7. Estructura del marco de componentes Lightning

Como vemos se apoya en Apex (lado del servidor) y JavaScript (lado del cliente). También puede contener otros componentes, así como HTML, CSS o cualquier otro código habilitado para la web. Estas clases pueden ser creadas en la consola de desarrollador y tienen distintas extensiones `.cmp` (Component), `.js` (Controller, Helper y Renderer), `.css` (Style), `.auradoc` (Documentation), `.design` (Design) y `.svg` (SVG). A continuación, se ve el ejemplo de “Hello World” al crear un componente aura y a la derecha todas las clases que lo forman.

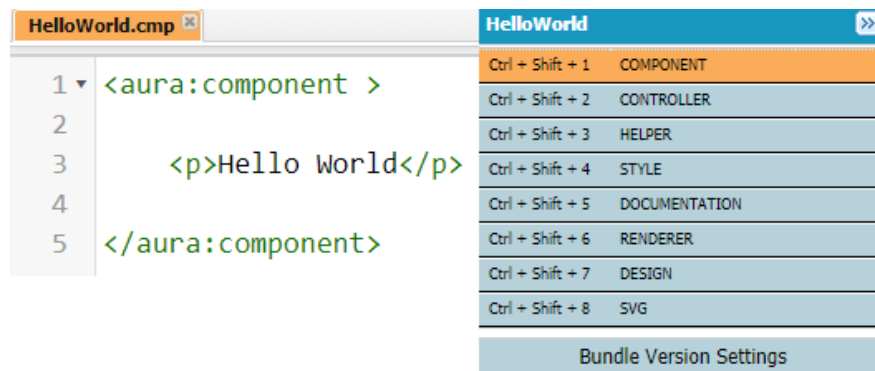


Figura 4-8. Ejemplo Hello World Aura

4.1.2.5 APIs de la Plataforma Lightning

Uno de los puntos clave de Salesforce es su integración con otras plataformas y lenguajes, es por eso que necesita de una serie de APIs que le permitan realizar operaciones entre cliente y servidor [\[18\]](#) [\[19\]](#). Las APIs de datos de Salesforce son:

- **SOAP API:** Es un servicio web eficiente y de gran capacidad basado en el protocolo estándar de SOAP, que trabaja solo con archivos XML.
- **REST API:** Es un servicio web sencillo y eficiente basado en los principios de RESTful, para ello se apoya en métodos HTTP.
- **Bulk API:** Especializada en la carga y consulta de grandes cantidades de datos a la vez.
- **Streaming API:** Especializada para configurar las notificaciones que se desencadenan cuando se realizan cambios en los datos.

Para este proyecto nos centraremos en el uso de la API REST debido a que no se van a cargar grandes cantidades de datos o en formato XML. Además, en caso de querer hacerlo se puede usar Data Loader para cargar estos datos masivamente mediante archivos csv.

4.1.3 Programación Visual

La programación visual se puede definir como la programación que utiliza más de una dimensión (textual), añadiendo la manipulación de elementos gráficos. Si nos ceñimos a la definición de este concepto, Salesforce no tiene herramientas 100% visuales, ya que en todas ellas hay que usar código, aunque sea de una forma mucho más compacta y

guiada. Por otro lado, esta programación visual está parcialmente basada en un paradigma declarativo, que ensalza la construcción de la estructura mediante elementos lógicos. Aunque, al contrario que en la programación declarativa, sí que hay que definir el flujo de control y ciertos algoritmos.

En definitiva, en este apartado llamaremos programación visual a toda herramienta que se aleje de la programación pura tradicional, ya sea aportando diagramas o elementos gráficos para el diseño de los distintos procesos [\[19\]](#).

4.1.3.1 Reglas de validación

Estas verifican que los datos ingresados por usuarios en registros cumplen las reglas que se especifican antes de poder guardarlos. Para esta verificación se escribe una expresión o fórmulas, en un lenguaje similar al pseudocódigo, con su propia biblioteca de funciones y con llamadas a los distintos campos del sistema. Esta expresión devuelve un valor booleano, si la casuística es true saltará la regla de validación impidiendo guardar el valor del campo al que se está aplicando esta regla, mientras que si es false la regla no saltará y permitirá guardar el valor. A continuación, se muestra un ejemplo de una regla de validación para que salte si no se cumple el formato asociado al DNI.

Formula de condición de error

Ejemplo: `Discount_Percent__c>0,30` [Más ejemplos...](#)
Mostrar un error si el descuento es superior al 30%

Si esta fórmula de expresión es **verdadera**, muestra el texto definido en el área del mensaje de error

`NOT (REGEX (NIF__c, "[0-9]{8} [A-Z]{1}"))`

Funciones
-- Todas las categorías de fi ▼
ABS
ADDMONTHS
AND
BEGINS
BLANKVALUE
BR

ABS(number)
Indica el valor absoluto de un número, un número sin signo

[Ayuda sobre esta función](#)

Figura 4-9. Ejemplo Regla de Validación

4.1.3.2 Process Builder

Es una herramienta interactiva que permite automatizar procesos de negocio y ver una representación gráfica del proceso. Para creación de un process builder hay que definir un desencadenador (por ejemplo, cuando se crea un candidato), después se establece un rombo de decisión (por ejemplo, el país del candidato es España) y por último se ejecutan una serie de acciones si se cumple este criterio. Estas acciones pueden ser inmediatas o programadas en el tiempo y abarcan actualizaciones de registros, envíos de emails, creación de nuevos registros, publicación de noticias, llamadas al código apex, etc. En la siguiente Figura se crea el ejemplo nombrado anteriormente, añadiendo la acción inmediata de enviar un mail de bienvenida y que tras 7 días, se actualice el estado del candidato.

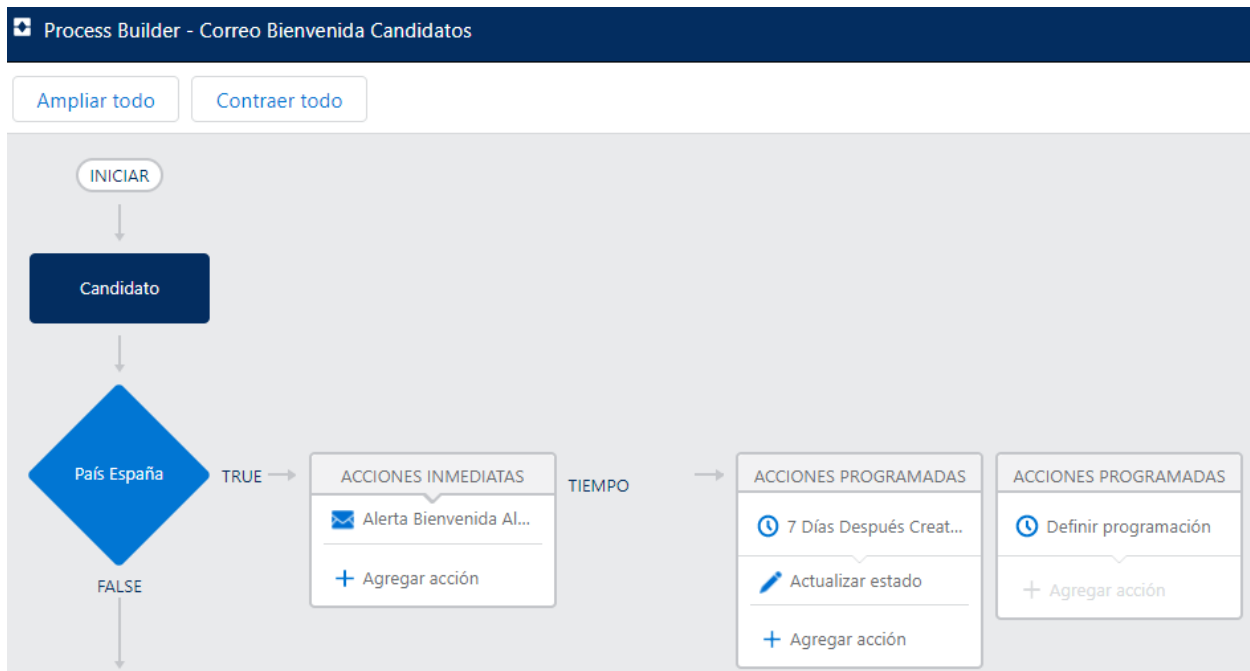


Figura 4-10. Ejemplo Process Builder

4.1.3.3 Workflows

Los flujos de trabajo permiten automatizar procedimientos y procesos estándar internos para ahorrar tiempo. Tienen una estructura muy similar a los process builders, incluso pueden ser parte de ellos.

Por último, un ejemplo de un flujo de trabajo simple con el mismo ejemplo que el realizado para el process builder en el apartado [4.1.3.2](#).

Detalle de Regla de flujo de trabajo

Modificar
 Eliminar
 Duplicar
 Activar

Nombre de regla	Envío Workflow Example	Objeto	Candidato
Activo	<input type="checkbox"/>	Criterios de evaluación	Evaluar la regla cuando un registro es creado
Descripción			
Criterios de reglas	Candidato: Estado de candidato IGUALA Working - Contacted		
Creado por	Javier López Quesada, 17/7/2021 18:27	Modificado por	Javier López Quesada, 17/7/2021 18:27

Acciones de flujo de trabajo

Modificar

Acciones de flujo de trabajo inmediatas

Tipo	Descripción
Tarea	Enviar correo

Acciones de flujo de trabajo dependientes del tiempo
[Ver un ejemplo](#)

	7 Días Después Fecha de activación de regla
Tipo	Descripción
Actualización de campo	Actualizar estado

Figura 4-11. Ejemplo Workflow

4.1.3.4 Flows

Es la herramienta que mejor describe el concepto de programación visual, ya que son declarativos, pero requieren que comprendas algunos conceptos de programación como variables y cómo funciona la lógica. Al igual que process builders y workflows, tienen una estructura similar, pero aportan muchas más opciones y elementos de programación. En base a cuando se disparan, existen 5 tipos de flujos:

1. Flujo de pantalla: Se inicia desde páginas Lightning, sitios de Experience Cloud o acciones rápidas.
2. Registro - Flujo desencadenado: Se inicia cuando se crea, se actualiza o se elimina un registro.
3. Programación - Flujo desencadenado: Se inicia a una hora y frecuencia especificadas para cada registro.
4. Evento de plataforma - Flujo desencadenado: Se inicia cuando se recibe un mensaje de evento de plataforma.
5. Flujo iniciado automáticamente (Sin desencadenador): Se inicia cuando se invoca mediante process builders, clases de Apex y otros procesos de Salesforce.

Todos los flujos, independientemente del tipo, son más completos que los process builder o los workflows ya que permiten entre otros usos:

- Crear nuevos recursos como variables, constantes, fórmulas, plantillas de texto o etapas.
- Llamar a todo tipo de acciones en Salesforce
- Recorrer listas de registro mediante bucles.
- Establecer nodos de decisión.

Por último, un ejemplo de un flujo simple con el mismo ejemplo que el realizado para el process builder en el apartado [4.1.3.2](#).

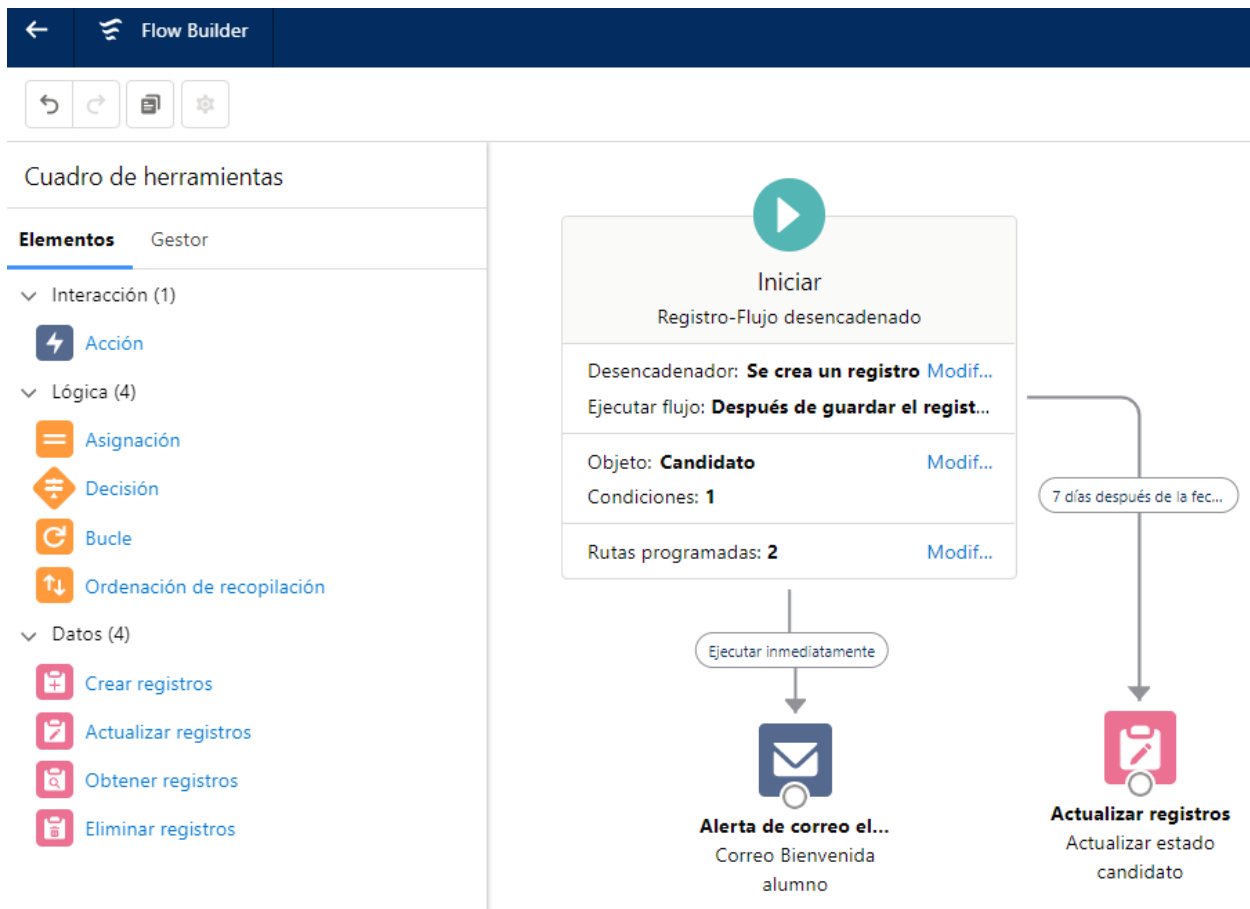


Figura 4-12. Ejemplo Flow

4.1.3.5 Reglas de duplicidad

Es la principal herramienta para evitar la duplicidad de campos, principalmente los estándar, ya que los personalizados permiten evitar la duplicidad marcándolo como único, impidiendo que este se repita. Para conseguir este cometido existen las reglas de coincidencia y las reglas duplicadas.

- **Reglas de coincidencia:** dado un tipo de objeto se determina el campo que tiene que coincidir para que salte la regla duplicada, alertando o impidiendo su creación o actualización.
- **Reglas duplicadas:** en ellas se determinan las acciones que se deben llevar a cabo una vez se cumplan las reglas de coincidencia a las que llama.

4.2 Conceptos gráficos

Una vez explicada la organización de los datos y de las automatizaciones sobre los mismos, hay que hacer hincapié en el apartado gráfico. Salesforce cuenta con varias herramientas para dar un mejor acabado a la interfaz que usará el usuario comercial. Las más usadas son los layouts, el generador de aplicaciones y, más enfocadas a la integración, las aplicaciones conectadas.

4.2.1 Formatos de página

Los page layouts o formatos de página permiten indicar qué campos, botones o acciones se mostrarán para los distintos objetos en función del tipo de registro, del usuario o de la aplicación que lo visualice. Gracias a ellos podemos resaltar los campos más interesantes y ocultar aquellos que no nos resulten útiles.

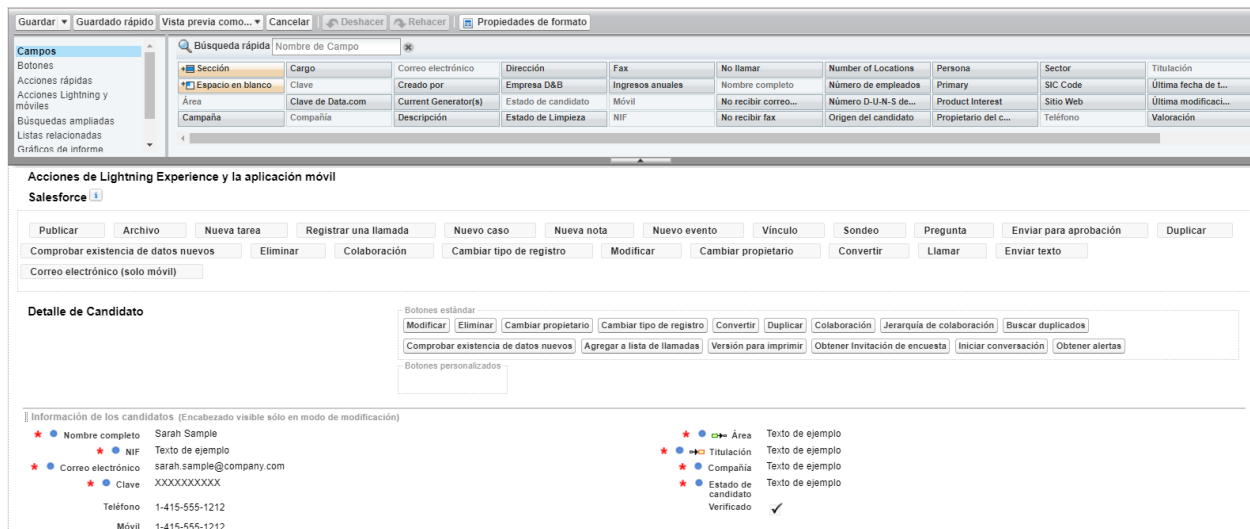


Figura 4-13. Ejemplo Editar Layout

Estos formatos de página son usados dentro de las páginas de registros Lightning que además permiten añadir hasta 32 nuevos componentes a la visualización principal.

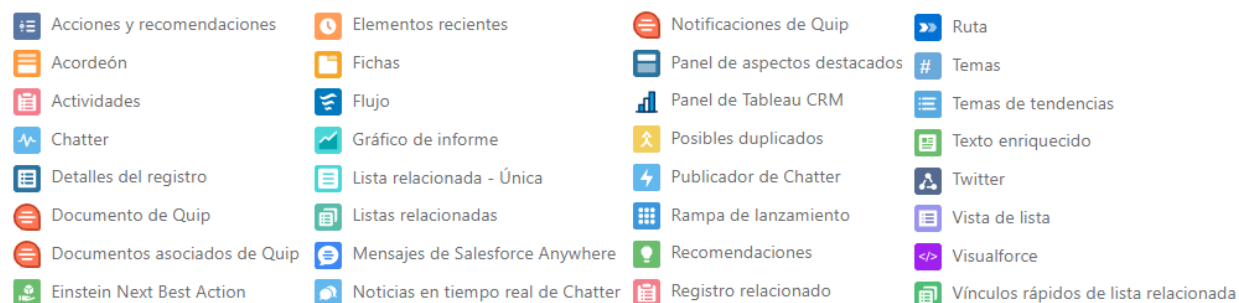


Figura 4-14. Componentes para páginas de registros Lightning

4.2.2 Gestor de aplicaciones

En Salesforce, una aplicación es un conjunto de elementos que funcionan juntos para atender una función concreta. Según su función, existen tres tipos de aplicaciones, clásicas, Lightning y conectadas.

4.2.2.1 Aplicaciones Clásicas

En sus inicios, Salesforce disponía de una interfaz gráfica llamada Salesforce Classic [\[19\]](#), pero con el paso de los años las necesidades de los clientes cambiaron y se vieron en la obligación de lanzar una nueva versión con componentes mucho más óptimos. A pesar de que ya prácticamente todos los usuarios se han pasado a esta nueva versión Lightning, existen algunas empresas más conservacionistas que siguen usando la versión clásica. Se podría decir que está actualmente en desuso, aunque cualquier proyecto puede ser cambiado a la versión clásica, pero visualmente es mucho más tediosa y no merece la pena ahondar más en ella.

4.2.2.2 Aplicaciones Lightning

La diferencia principal con las aplicaciones clásicas es la inclusión de elementos Lightning en la misma, listados en la *figura 4-14*. Por este motivo, es mucho mejor crear una aplicación de este tipo, ya que sólo aportan funcionalidad extra y no eliminan nada de la versión clásica. Además, el modelo de navegación está mucho más optimizado, siendo notablemente rápido e intuitivo.

4.2.2.3 Aplicaciones Conectadas

Una aplicación conectada es un marco de trabajo que activa una aplicación externa para la integración con Salesforce utilizando protocolos estándar y API, como SAML, OAuth y OpenID Connect. Esta conexión se realiza a través de un client ID y su client secret, con los cuales se genera un token de acceso para autorizarse en la integración. Además, las aplicaciones conectadas permiten múltiples gestiones entre las que se encuentran:

- Limitar la información a la que se puede acceder, a través de la configuración de políticas del protocolo.
- Permitir la inclusión de certificados o claves digitales que impiden accesos indebidos.
- Añadir un conjunto de direcciones IP de confianza.
- Limitar por perfiles y conjuntos de permisos.
- Gestionar el protocolo CORS para el intercambio de recursos.

Capítulo 5 - Desarrollo del prototipo

En este capítulo se describe todo el proceso de desarrollo del prototipo, desde el método seguido, pasando por la funcionalidad y finalizando con un caso práctico de todo el proceso.

5.1 Marco de trabajo

Para este proyecto el modelo más adecuado sería Scrum, sin embargo, este modelo sería útil para un desarrollo completo de la aplicación y como el objetivo del proyecto es la creación de un prototipo, usaremos el modelo de prototipos (modelo de desarrollo evolutivo que construye un prototipo de forma sencilla, en poco tiempo y sin utilizar una gran cantidad de recursos). Se ha elegido este modelo por las siguientes razones:

1. Se conocen los objetivos generales de la aplicación, como son la correcta adaptación del concepto de CRM a un gestor de prácticas, pero no se conocen los requisitos de forma detallada.
2. Se necesita una continua retroalimentación por parte de los clientes o usuarios (alumno, empresa y coordinador de prácticas), de esta forma se añaden y modifican los requisitos en base a lo acordado. Es decir, el prototipo se ajusta para satisfacer las necesidades del cliente/usuario.
3. Existe cierta inseguridad por parte del desarrollador en la usabilidad de la aplicación y se necesita de un prototipo para mejorarla en base a este y a las pruebas realizadas sobre él.
4. Se dispone de un menor tiempo dirigido al desarrollo (5 meses) debido a que está más orientado a la investigación. Por tanto, la construcción del prototipo debe ser realizada en poco tiempo.
5. Se disponen de pocos recursos para el desarrollo (un solo programador y un único equipo de prestaciones medias).
6. Se puede reusar código y procedimientos estandarizados, ya que esto no tiene porqué aparecer en la versión final, pero si ahorrará tiempo y trabajo para la realización de pruebas.

5.2 Adaptación al gestor de prácticas PSUCM

En este punto se tratará toda la concepción y adaptación de un sistema CRM a un gestor de prácticas. PSUCM (Prácticas Salesforce Universidad Complutense de Madrid) es una plataforma de prácticas que trata de sacar partido a las distintas funcionalidades de Salesforce, para proponer una alternativa a la actual plataforma de la Universidad Complutense de Madrid (GIPE). Para llevar a cabo este proceso, hay que tener en cuenta múltiples factores que van desde la funcionalidad del sistema, la creación de objetos y del flujo de trabajo, hasta el diseño y la integración de la propia aplicación.

5.2.1 Procedimiento y concepción previa de PSUCM

Antes de comenzar con la definición de objetos, el diseño de la aplicación o la integración es necesario introducir como es el funcionamiento de un gestor de prácticas y que actores interactúan con el mismo. Para conseguir esta información se ha hecho una entrevista a Daniel Ángel Chaver Martínez, Vicedecano de Relaciones Externas e Investigación, cuya función principal es la de coordinar todas las prácticas externas, realizadas en la facultad de informática, a través de la plataforma del GIPE. En esta entrevista se le pregunta por aspectos relacionados con el procedimiento llevado a cabo en la asignación de prácticas curriculares, así como los aspectos más destacados y también las principales carencias de la actual plataforma de prácticas. La entrevista íntegra se puede visualizar en el [\[Apéndice A\]](#), aunque a continuación se extraerán las ideas principales para sacar una serie de conclusiones a la hora del desarrollo de este proyecto.

En lo referente al procedimiento, Daniel destacó la unificación de la plataforma entre alumnos, empresas y el propio coordinador de prácticas, destacando el hecho de que todo se gestiona desde la misma plataforma y explicando todo el proceso con sus distintas interacciones y documentos clave. Este proceso se podría resumir con el siguiente esquema.

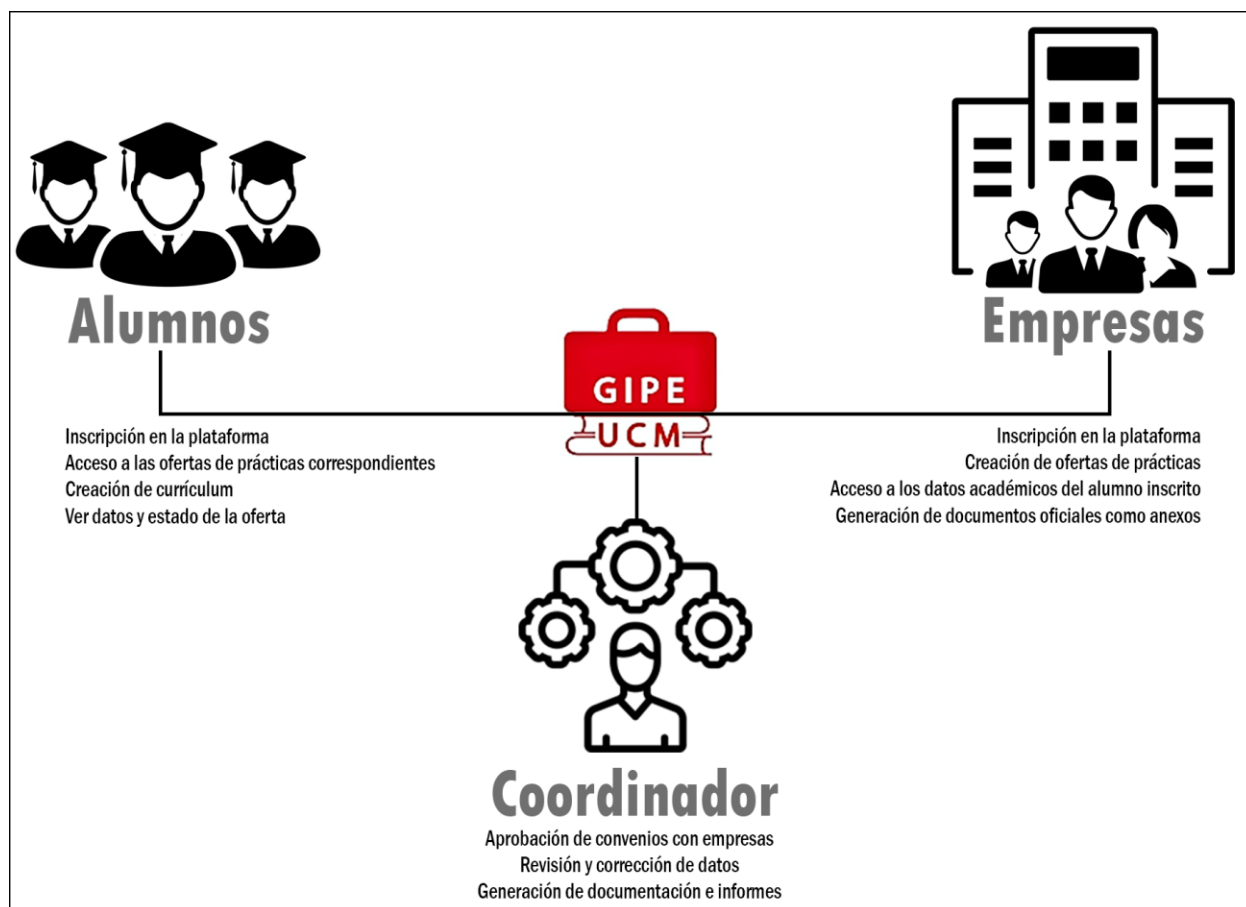


Figura 5-1. Esquema estructural del proceso de gestión de prácticas

Como principales carencias de la plataforma señaló la falta de sincronización con el GEA, el portal donde se guardan los datos académicos del alumno, y también el hecho de que para acceder a cada alumno hay que hacerlo en base a la titulación, no pudiendo acceder a través de una vista general que permita verlos todos. Además, apunta que muchos elementos del GIPE no son intuitivos y a los usuarios menos experimentados les cuesta moverse por el mismo. En contraposición con estas posibles mejoras, destacó los aspectos positivos de la plataforma además de la unificación ya comentada, destaca las automatizaciones a la hora de generar anexos o informes.

En cuanto a las nuevas tecnologías que incluiría, comentó que para el tema de los contratos sería muy interesante la inclusión del Blockchain, aunque no lo ve tan indispensable como la incorporación de una plataforma de firma digital, ya que en el GIPE se sigue incluyendo la firma de forma manual. No obstante, los procesos recogidos

en la plataforma no le parecen lo suficientemente complejos para meter otras herramientas relacionadas con la inteligencia artificial, aunque no se cierra a ideas innovadoras o que mejoren la plataforma a nivel de eficiencia.

A nivel de coordinación de la plataforma, Daniel no puede modificar la plataforma de prácticas, aunque sí que dispone de un equipo de soporte al que puede comentar fallos y posibles mejoras, aunque también reconoció que para universidades con menos recursos sería muy interesante que este proceso corriera a cargo del gestor de prácticas y así abaratar costes. Aunque destaca la dificultad para usuarios no técnicos en el manejo de las plataformas, por lo que las automatizaciones habría que hacerlas preferiblemente con opciones de configuración y programación visual, detallándolo muy bien en un manual de uso dirigido a coordinadores.

Basándonos en la información aportada durante la entrevista, se pueden sacar los siguientes puntos clave sobre aspectos a tener en cuenta en la aplicación de este proyecto.

- Estructura basada en alumnos, empresas y coordinador de prácticas.
- Sincronización de datos con el GEA.
- Distintas formas de visualización de alumnos y empresas por parte del coordinador.
- Diseño intuitivo y creación de un manual de uso.
- Contratos apoyados en la tecnología Blockchain.
- Plataforma de firma electrónica integrada.
- Generación automática de correos, informes y anexos.
- Mayor protagonismo de elementos configurables y de programación visual, por encima de la programación más purista.

5.2.2 Análisis de requisitos de PSUCM

Existen dos tipos de requisitos, los funcionales que describen lo que hace un sistema o lo que se espera que haga y los no funcionales que describen aspectos del sistema para poder cumplir los funcionales. Debido a que en este proyecto existen dos aplicaciones, un web-service para alumnos/empresas y un entorno para el coordinador de prácticas

en Salesforce, se establecerán dos nomenclaturas distintas. RFWXX (Requisito funcional del web-service número XX), RFSXX (Requisito funcional de Salesforce número XX), RNFWXX (Requisito no funcional del web-service número XX) y RNFSXX (Requisito no funcional de Salesforce número XX). De igual forma en el diagrama de casos de uso existen dos contenedores para cada aplicación (Figura 5-2).

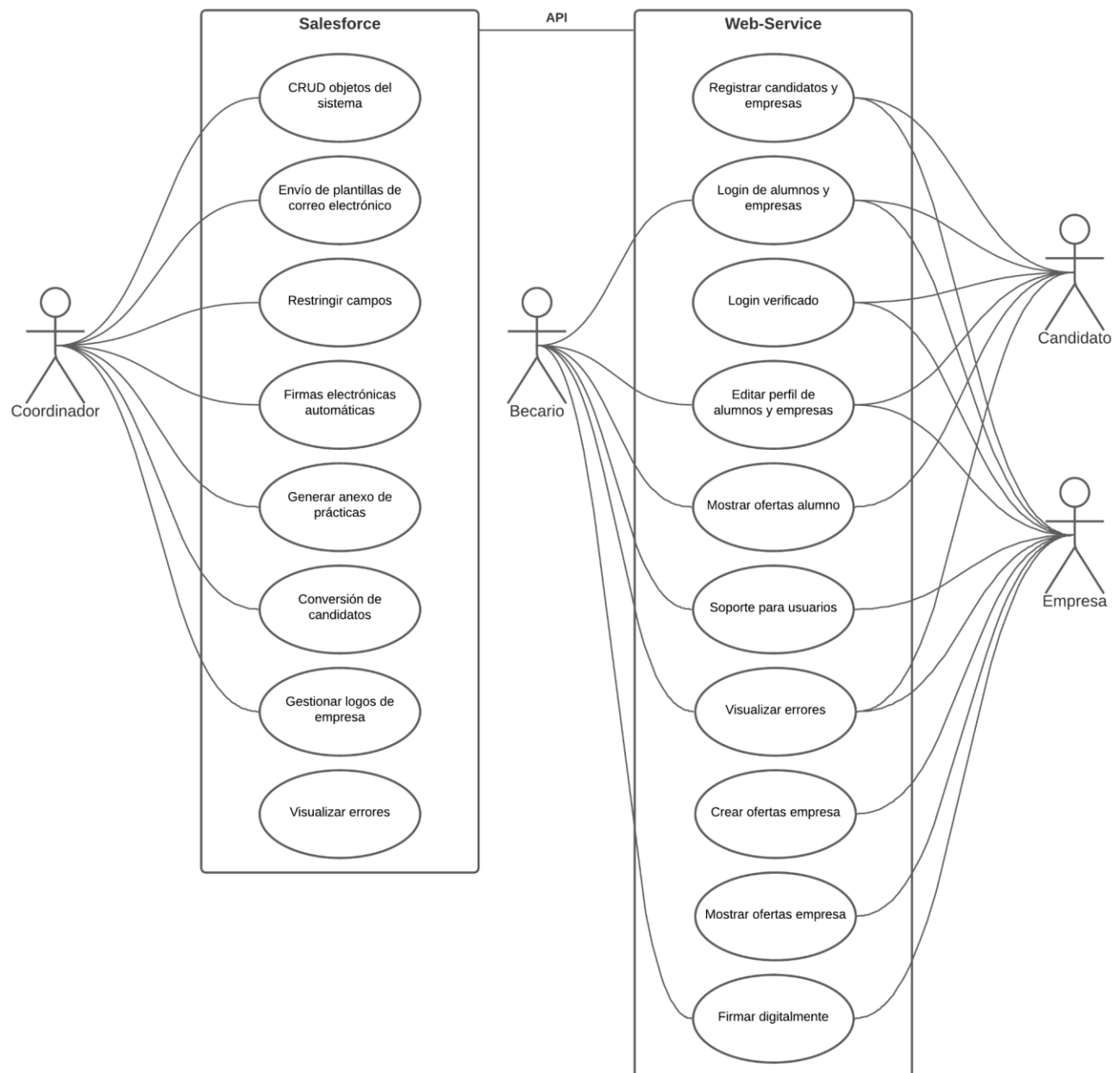


Figura 5-2. Diagrama de casos de uso funcional

Los **requisitos funcionales** de este proyecto son:

- Registrar candidatos y empresas.

Nº	ATRIBUTO	CONTENIDO
1	Identificador	RFW01
2	Nombre	Registrar candidatos y empresas
3	Versión	1.0
4	Prioridad	Alta
5	Descripción	Debe tener un registro para alumnos candidatos a becarios y otro registro para empresas que deseen publicar ofertas. Estos formularios tendrán una serie de campos seleccionados.

Tabla 5-1. Requisito funcional Web 1

- Login de alumnos y empresas.

Nº	ATRIBUTO	CONTENIDO
1	Identificador	RFW02
2	Nombre	Login de alumnos y empresas
3	Versión	1.0
4	Prioridad	Alta
5	Descripción	Debe tener un login para los alumnos tanto si son candidatos a becarios como si ya figuran en el sistema como becarios. Del mismo modo existirá un login para el acceso de empresas.

Tabla 5-2. Requisito funcional Web 2

- Login verificado.

Nº	ATRIBUTO	CONTENIDO
1	Identificador	RFW03
2	Nombre	Login verificado
3	Versión	1.0
4	Prioridad	Media

5	Descripción	No se podrá hacer login mientras que el coordinador de prácticas no marque el campo de verificación en el alumno o empresa correspondiente.
---	-------------	---

Tabla 5-3. Requisito funcional Web 3

- Editar perfil de alumnos y empresas.

Nº	ATRIBUTO	CONTENIDO
1	Identificador	RFW04
2	Nombre	Editar perfil de alumnos y empresas
3	Versión	1.0
4	Prioridad	Media
5	Descripción	Sección reservada a la edición del perfil de alumnos candidatos y becarios. Del mismo modo existirá un perfil para la edición de datos por parte de las empresas.

Tabla 5-4. Requisito funcional Web 4

- Crear ofertas empresa.

Nº	ATRIBUTO	CONTENIDO
1	Identificador	RFW05
2	Nombre	Crear ofertas empresa
3	Versión	1.0
4	Prioridad	Alta
5	Descripción	Las empresas podrán crear ofertas de trabajo asociadas a su propia entidad para que los candidatos puedan inscribirse en ellas.

Tabla 5-5. Requisito funcional Web 5

- Mostrar ofertas empresa.

Nº	ATRIBUTO	CONTENIDO
1	Identificador	RFW05
2	Nombre	Mostrar ofertas empresa
3	Versión	1.0
4	Prioridad	Alta
5	Descripción	La empresa podrá visualizar todas las ofertas creadas por la misma, así como todos los usuarios que se inscriban en ella.

Tabla 5-6. Requisito funcional Web 6

- Mostrar ofertas alumno.

Nº	ATRIBUTO	CONTENIDO
1	Identificador	RFW07
2	Nombre	Mostrar ofertas alumno
3	Versión	1.0
4	Prioridad	Alta
5	Descripción	Los alumnos podrán visualizar todas las ofertas creadas por empresas, siempre y cuando pertenezcan a su área. Además podrá inscribirse o desinscribirse de las mismas.

Tabla 5-7. Requisito funcional Web 7

- Soporte para usuarios.

Nº	ATRIBUTO	CONTENIDO
1	Identificador	RFW08
2	Nombre	Soporte para usuarios
3	Versión	1.0
4	Prioridad	Media
5	Descripción	Tanto empresas como alumnos tendrán la opción de abrir casos para notificar incidencias o exponer problemas durante

		el transcurso de las prácticas.
--	--	---------------------------------

Tabla 5-8. Requisito funcional Web 8

- Visualizar errores.

Nº	ATRIBUTO	CONTENIDO
1	Identificador	RFW09
2	Nombre	Visualizar errores
3	Versión	1.0
4	Prioridad	Alta
5	Descripción	En todas las vistas de la aplicación se mostrarán los errores definidos en Salesforce (restricciones de campos, duplicidad de campos clave y obligatoriedad de los mismos).

Tabla 5-9. Requisito funcional Web 9

- Firmar digitalmente.

Nº	ATRIBUTO	CONTENIDO
1	Identificador	RFW10
2	Nombre	Firmar digitalmente
3	Versión	1.0
4	Prioridad	Alta
5	Descripción	La página tendrá un sistema para que becarios y empresas puedan firmar digitalmente.

Tabla 5-10. Requisito funcional Web 10

- CRUD objetos del sistema.

Nº	ATRIBUTO	CONTENIDO
1	Identificador	RFS01
2	Nombre	CRUD objetos del sistema
3	Versión	1.0

4	Prioridad	Alta
5	Descripción	El coordinador de prácticas podrá crear, leer, actualizar y borrar los objetos definidos en el sistema manualmente (candidatos, becarios, empresas, ofertas, casos, firmas, informes y archivos). Para ello se añadirán los botones correspondientes en las distintas fichas de cada objeto.

Tabla 5-11. Requisito funcional Salesforce 1

- Envío de plantillas de correo electrónico.

Nº	ATRIBUTO	CONTENIDO
1	Identificador	RFS02
2	Nombre	Envío de plantillas de correo electrónico.
3	Versión	1.0
4	Prioridad	Alta
5	Descripción	El sistema enviará automáticamente: - Plantillas de bienvenida al crear candidatos y cuentas. - Aviso de verificación cuando el gestor marque el campo correspondiente. - Plantilla de conversión de alumno candidato a alumno becario.

Tabla 5-12. Requisito funcional Salesforce 2

- Restringir campos

Nº	ATRIBUTO	CONTENIDO
1	Identificador	RFS03
2	Nombre	Restringir campos.
3	Versión	1.0
4	Prioridad	Alta
5	Descripción	Ciertos campos del sistema tendrán limitaciones especificadas en el punto 5.2.3 y el gestor podrá modificarlas.

Tabla 5-13. Requisito funcional Salesforce 3

- Firmas electrónicas automáticas.

Nº	ATRIBUTO	CONTENIDO
1	Identificador	RFS04
2	Nombre	Firmas electrónicas automáticas.
3	Versión	1.0
4	Prioridad	Media
5	Descripción	El sistema permitirá registrar firmas a contactos y empresas automáticamente.

Tabla 5-14. Requisito funcional Salesforce 4

- Generar anexo de prácticas.

Nº	ATRIBUTO	CONTENIDO
1	Identificador	RFS05
2	Nombre	Generar anexo de prácticas
3	Versión	1.0
4	Prioridad	Media
5	Descripción	El sistema generará automáticamente una página de visualforce en formato pdf que incluye las firmas de la empresa, el alumno y detalles de la oferta.

Tabla 5-15. Requisito funcional Salesforce 5

- Conversión de candidatos.

Nº	ATRIBUTO	CONTENIDO
1	Identificador	RFS06
2	Nombre	Conversión de candidatos.
3	Versión	1.0
4	Prioridad	Alta
5	Descripción	El sistema permitirá realizar una conversión de candidatos a becarios. Para ello en el proceso se elegirá una empresa del

		sistema y una de las ofertas que pertenecen a la misma.
--	--	---

Tabla 5-16. Requisito funcional Salesforce 6

- Gestionar logos de empresa.

Nº	ATRIBUTO	CONTENIDO
1	Identificador	RFS07
2	Nombre	Gestionar logos de empresa.
3	Versión	1.0
4	Prioridad	Media
5	Descripción	Los logos de la empresa se almacenarán en el sistema de manera que en base al número de identificación de la empresa y se asignarán automáticamente una vez creada o actualizada la misma.

Tabla 5-17. Requisito funcional Salesforce 7

- Visualizar errores.

Nº	ATRIBUTO	CONTENIDO
1	Identificador	RFS08
2	Nombre	Visualizar errores
3	Versión	1.0
4	Prioridad	Alta
5	Descripción	En todas las vistas de la aplicación se mostrarán los errores definidos para evitar errores (restricciones de campos, duplicidad de campos clave y obligatoriedad de los mismos).

Tabla 5-18. Requisito funcional Salesforce 8

Por otro lado, los **requisitos no funcionales** de este proyecto son:

- Acceso a internet

Nº	ATRIBUTO	CONTENIDO
1	Identificador	RNFW01
2	Nombre	Acceso a internet
3	Versión	1.0
4	Prioridad	Alta
5	Descripción	Es necesario disponer de acceso a internet para la aplicación web, ya que usa multitud de recursos alojados en la nube propios de Salesforce.

Tabla 5-19. Requisito no funcional Web 1

- Back-end operativo

Nº	ATRIBUTO	CONTENIDO
1	Identificador	RNFW02
2	Nombre	Back-end operativo
3	Versión	1.0
4	Prioridad	Alta
5	Descripción	Para la correcta utilización del web-service (Angular) requiere que el back-end (Salesforce) funcione correctamente, ya que se hacen continuas peticiones al mismo.

Tabla 5-20. Requisito no funcional Web 2

- Uso multiplataforma

Nº	ATRIBUTO	CONTENIDO
1	Identificador	RNFW03
2	Nombre	Uso multiplataforma
3	Versión	1.0
4	Prioridad	Alta

5	Descripción	La aplicación parte front del webservice será programada en Angular para su uso en PC. Aunque también se podría visualizar en dispositivos móviles, no se garantiza la armonía entre los distintos elementos que la componen.
---	-------------	---

Tabla 5-21. Requisito no funcional Web 3

- Protección de datos

Nº	ATRIBUTO	CONTENIDO
1	Identificador	RNFW04
2	Nombre	Protección de datos
3	Versión	1.0
4	Prioridad	Alta
5	Descripción	Debe cumplir todos los requisitos de seguridad y protección de datos estipulados por el BOE en la Ley Orgánica 3/2018, del 5 de diciembre, de Protección de Datos Personales y garantía de los derechos digitales.

Tabla 5-22. Requisito no funcional Web 4

- Token de seguridad

Nº	ATRIBUTO	CONTENIDO
1	Identificador	RNFW05
2	Nombre	Token de seguridad
3	Versión	1.0
4	Prioridad	Alta
5	Descripción	El token de seguridad que da acceso a Salesforce debe ser generado externamente con Postman, debido a los bloqueadores CORS que garantizan la seguridad del propio CRM.

Tabla 5-23. Requisito no funcional Web 5

- Acceso a internet

Nº	ATRIBUTO	CONTENIDO
1	Identificador	RNFS01
2	Nombre	Acceso a internet
3	Versión	1.0
4	Prioridad	Alta
5	Descripción	Es necesario disponer de acceso a internet para poder acceder a Salesforce, ya que usa multitud de recursos alojados en la nube.

Tabla 5-24. Requisito no funcional Salesforce 1

- Bugs de Salesforce

Nº	ATRIBUTO	CONTENIDO
1	Identificador	RNFS02
2	Nombre	Bugs de Salesforce
3	Versión	1.0
4	Prioridad	Alta
5	Descripción	La aplicación está sujeta al correcto funcionamiento de Salesforce, si este presenta algún bug interno afectará a la aplicación hasta que se solucione

Tabla 5-25. Requisito no funcional Salesforce 2

- Uso multiplataforma

Nº	ATRIBUTO	CONTENIDO
1	Identificador	RNFS03
2	Nombre	Uso multiplataforma
3	Versión	1.0
4	Prioridad	Alta
5	Descripción	La parte front que corresponde a Salesforce, es compatible

		tanto en PC como en Smartphones u otros dispositivos inteligentes.
--	--	--

Tabla 5-26. Requisito no funcional Salesforce 3

- Protección de datos

Nº	ATRIBUTO	CONTENIDO
1	Identificador	RNFS04
2	Nombre	Protección de datos
3	Versión	1.0
4	Prioridad	Alta
5	Descripción	Debe cumplir todos los requisitos de seguridad y protección de datos estipulados por el BOE en la Ley Orgánica 3/2018, del 5 de diciembre, de Protección de Datos Personales y garantía de los derechos digitales.

Tabla 5-27. Requisito no funcional Salesforce 4

5.2.3 Objetos y campos de PSUCM

Para desarrollar el prototipo mencionado anteriormente es necesario crear y adaptar en Salesforce una serie de objetos, campos y relaciones que sirvan de base para la creación de todo el flujo de funcionalidades de esta aplicación. La prioridad es usar el mayor número de objetos y campos estándar ya definidos (*Figura 5-2*), ya que sería la forma correcta de adaptar el concepto de un CRM a todo este proceso académico. Sin embargo, esto es muy complicado ya que hay mucha información que un cliente y un alumno no comparten. Por tanto, para el diseño del sistema se han usado objetos estándar renombrados, añadiendo campos personalizados y relaciones entre los propios objetos. De esta forma el diagrama UML quedaría de la siguiente manera (*Figura 5-3*).

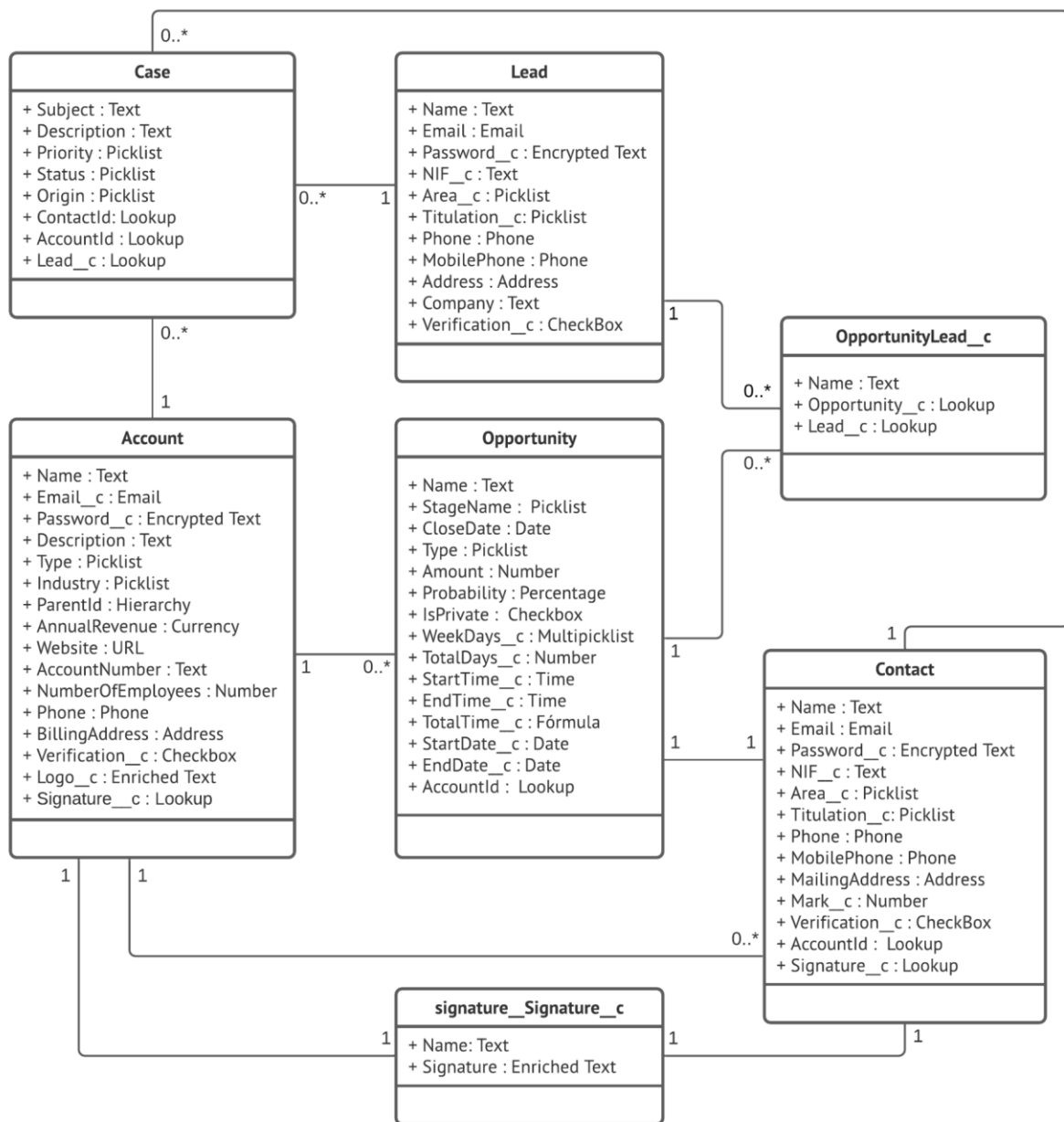


Figura 5-3. Diagrama de clases Salesforce

A continuación, se explica qué uso se le da a los principales campos del sistema, así como su nombre, nombre en la api, restricciones y obligatoriedad del mismo. Los objetos que se usarán para la realización del prototipo son:

Candidato / Lead: Objeto estándar que representará al alumno que quiere optar a unas prácticas. El mero registro del alumno en la plataforma front de PSUCM supondrá la

creación de un registro en dicho objeto, aunque también puede ser creado manualmente por el gestor de prácticas en Salesforce. Los campos usados son:

Nombre	Descripción	Tipo y restricciones	Obligatorio
Nombre / Name	Nombre completo del alumno candidato	Texto hasta 255 caracteres	Sí
Correo / Email	Correo electrónico del alumno candidato	Correo con extensión '@ucm.es'	Sí
Clave / Password__c	Clave cifrada para poder acceder desde el web service	Texto cifrado entre 8 y 26 caracteres	Sí
NIF / NIF__c	Número de identificación fiscal del alumno candidato	Texto con 8 caracteres numéricos y 1 letra	Sí
Área / Area__c	Lista de selección sobre el área a la que pertenece el alumno candidato (Artes y Humanidades, Ciencias Sociales y Jurídicas, Ciencias, Ciencias de la Salud o Ingenierías y Arquitectura)	Lista de selección única con valores seleccionados	Sí
Titulación / Titulation__c	Lista de selección sobre la titulación a la que pertenece el alumno candidato (Grado en Ingeniería Informática, Grado en Periodismo, Grado en Enfermería, etc.)	Lista de selección única con valores seleccionados dependientes del área seleccionada	Sí
Teléfono / Phone	Teléfono fijo del alumno candidato	Teléfono con un máximo de 40 dígitos	No
Móvil / MobilePhone	Teléfono móvil del alumno candidato	Teléfono con un máximo de 40 dígitos	No
Universidad / Company	Universidad de la que proviene el alumno candidato	Texto con valor constante "Universidad Complutense de Madrid"	Sí
Dirección / Address	Dirección de correo postal del alumno candidato	Dirección formada por varios campos; Calle, Código postal, Ciudad, Estado o provincia y país de facturación	No

Verificado / Verification__c	Casilla utilizada para saber si un candidato ha sido verificado por el gestor de prácticas.	Casilla	Sí
---------------------------------	---	---------	----

Tabla 5-28. Campos del objeto Candidato

Becario / Contact: Objeto estándar (renombrado) que representará al alumno una vez lo hayan elegido en la oferta de prácticas.

Nombre / Api	Descripción	Tipo y Restricciones	Obligatorio
Nombre / Name	Nombre completo del alumno becario	Texto hasta 255 caracteres	Sí
Correo / Email	Correo electrónico del alumno becario	Correo con extensión '@ucm.es'	Sí
Clave / Password__c	Clave cifrada para poder acceder desde el web service	Texto cifrado entre 8 y 26 caracteres	Sí
NIF / NIF__c	Número de identificación fiscal del alumno becario	Texto con 8 caracteres numéricos y 1 letra	Sí
Área / Area__c	Lista de selección sobre el área a la que pertenece el alumno becario (Artes y Humanidades, Ciencias Sociales y Jurídicas, Ciencias, Ciencias de la Salud o Ingenierías y Arquitectura)	Lista de selección única con valores seleccionados	Sí
Titulación / Titulation__c	Lista de selección sobre la titulación a la que pertenece el alumno becario (Grado en Ingeniería Informática, Grado en Periodismo, Grado en Enfermería, etc.)	Lista de selección única con valores seleccionados dependientes del área seleccionada	Sí
Teléfono / Phone	Teléfono fijo del alumno candidato	Teléfono con un máximo de 40 dígitos	No
Móvil / MobilePhone	Teléfono móvil del alumno candidato	Teléfono con un máximo de 40 dígitos	No
Dirección de correo / mailingAddress	Dirección de correo postal del alumno becario	Dirección formada por varios campos; Calle, Código postal, Ciudad, Estado o provincia y País de facturación	No

Nota / Mark__c	Nota del alumno becario al finalizar las prácticas de empresa	Número entre 0 y 10	No
Verificado / Verification__c	Casilla utilizada para saber si un contacto ha sido verificado por el gestor de prácticas	Casilla	Sí
Nombre de la empresa / AccountId	Empresa relacionada con el contacto debido a que es la que le proporciona la práctica	Lookup a cuenta	Sí

Tabla 5-29. Campos del objeto Becario

Empresa / Account: Objeto estándar (renombrado) que representará la empresa que podrá ofrecer sus ofertas de trabajo.

Nombre / Api	Descripción	Tipo y Restricciones	Obligatorio
Nombre de la empresa / Name	Nombre completo del alumno	Texto hasta 255 caracteres	Sí
Correo / Email__c	Correo electrónico de la empresa	Correo electrónico	Sí
Clave / Password__c	Campo cifrado para poder acceder desde el web service	Texto cifrado entre 8 y 26 caracteres	Sí
Descripción / Description	Breve descripción de la empresa	Texto hasta 255 caracteres	Sí
Tipo / Type	Indica el tipo de empresa (Empresa individual, Sociedad Limitada, Sociedad Anónima, Sociedad Cooperativa o Start-up)	Lista de selección única con valores seleccionados	Sí
Sector / Industry	Indica el sector al que pertenece la empresa (Agrícola, industrial, ganadero, etc.)	Lista de selección única con valores seleccionados	Sí
Empresa principal / ParentId	Empresa a la que pertenece, si la hubiera.	ID Jerárquico	No
Ingresos anuales / AnnualRevenue	Ingresos estimados ganados por la empresa durante el último año	Divisa hasta 18 cifras	No

Sitio web / Website	Página web de la empresa	URL hasta 255 caracteres	No
Número de la empresa / AccountNumber	Número de identificación de la empresa	Texto hasta 40 caracteres	Sí
Empleados / NumberOfEmployees	Número de empleados que posee la empresa actualmente	Número hasta 8 cifras	No
Teléfono / Phone	Teléfono de contacto de la empresa	Teléfono	No
Dirección de facturación / BillingAddress	Dirección para la facturación de la empresa	Dirección formada por varios campos; Calle, Código postal, Ciudad, Estado o provincia y País de facturación	No
Verificada / Verification__c	Casilla utilizada para saber si una cuenta ha sido verificada por el gestor de prácticas	Casilla	Sí
Logo / Logo__c	Logo o marca de la empresa	Texto enriquecido	No

Tabla 5-30. Campos del objeto Empresa

Oferta / Opportunity: Objeto estándar que representará cada una de las ofertas emitidas por la empresa.

Nombre / Api	Descripción	Restricciones	Obligatorio
Nombre de la oferta/ Name	Nombre de la oferta de trabajo	Texto hasta 120 caracteres	No
Nombre de la empresa / AccountId	Empresa que lanza la oferta de trabajo	Lookup cuenta existente	No
Etapas / StageName	Etapas de la oferta de trabajo (Sin asignar, Asignada, En curso,	Lista de selección única	Sí

	Evaluando, Aprobado o Suspenso)		
Fecha de cierre / CloseDate	Fecha de finalización de la oferta de trabajo	Fecha	Sí
Tipo / Type	Tipo de oferta (Prácticas curriculares, Prácticas extracurriculares o Ambas)	Lista de selección única	No
Importe / Amount	Ingreso bruto mensual otorgado durante la realización de las prácticas	Divisa hasta 16 cifras y con dos cifras decimales	Sí
Probabilidad / Probability	Porcentaje ligado a la etapa, cuanto más alto más próximo a la finalización de las prácticas	Porcentaje de hasta tres cifras	Sí
Privada / IsPrivate	Determina si los candidatos pueden ver la oferta o sólo lo hará el gestor de prácticas	Casilla	No
Días de la semana / WeekDays__c	Días de la semana en los que se realizarán las prácticas (Lunes, martes, miércoles, jueves, viernes, sábado, domingo)	Lista de selección múltiple	Sí
Total de Días / TotalDays__c	Duración del periodo de prácticas en número de días, sin incluir fines de semana y festivos	Número entero de hasta 3 cifras	Sí
Hora de Inicio / StartTime__c	Hora de fin de la jornada	Hora	Sí
Hora de Fin / EndTime__c	Hora de fin de la jornada	Hora	Sí
Total de horas / TotalTime__c	Total de horas de la oferta de prácticas	Fórmula que devuelve un valor numérico (EndTime - StartTime) * TotalDays	Sí
Fecha de Inicio /	Fecha en la que comienza la práctica ofertada	Fecha	Sí

StartDate__c			
Fecha de Fin / EndDate__c	Fecha en la que finaliza la práctica ofertada	Fecha	Sí
Nombre de la empresa / AccountId	Empresa a la que pertenece la oportunidad debido a que es la que le proporciona la	Lookup a cuenta	No
Becario / ContactId	Contacto relacionada con el contacto debido a que es la que le proporciona la	Lookup a contacto	No

Tabla 5-31. Campos del objeto Oferta

Oferta-Candidato / OpportunityLead__c: Objeto cruzado personalizado que será usado para establecer una relación muchos a muchos entre ofertas y candidatos, de manera que un candidato puede tener muchas ofertas en las que esté inscrito y una oferta puede tener muchos candidatos.

Nombre / Api	Descripción	Restricciones	Obligatorio
Nombre de OfertaCandidato / Name	Nombre generado automáticamente	Texto autonumérico	Sí
Oferta / Opportunity__c	Oferta asociada al objeto	Lookup a oferta	Sí
Candidato / Lead__c	Candidato asociado al objeto	Lookup a candidato	Sí

Tabla 5-32. Campos del objeto Oferta-Candidato

Caso / Case: Objeto estándar que será usado para reportar incidencias por parte de candidatos, becarios o empresas.

Nombre / Api	Descripción	Restricciones	Obligatorio
Número del caso / CaseNumber	Número del caso generado automáticamente	Texto autonumérico	Sí
Nombre del candidato / Lead__c	Candidato asociado al caso	Lookup a candidato	No

Nombre del contacto / ContactId	Becario asociado al caso	Lookup a becario	No
Nombre de la empresa / AccountId	Empresa asociada al caso	Lookup a empresa	No
Estado / Status	Estado del caso (nuevo, trabajando o escalado)	Lista de selección	Sí
Asunto / Subject	Asunto del caso, donde se resumirá brevemente el cometido del mismo	Texto de hasta 255 caracteres	No
Descripción / Description	Descripción del caso, donde se especifican los detalles del mismo	Texto de hasta 32000 caracteres	No
Prioridad / Priority	Prioridad del caso (alta, media o baja)	Lista de selección	No
Origen / Origin	Origen del caso (Teléfono, email o web)	Lista de selección	Sí

Tabla 5-33. Campos del objeto Caso

5.2.4 Aplicaciones de PSUCM

Para la visualización en Salesforce se han creado dos apps con el gestor de aplicación con dos utilidades distintas. La aplicación Lightning tiene como finalidad mejorar la visualización de objetos de cara al coordinador de prácticas, mientras que la aplicación conectada busca la integración de la instancia de Salesforce con otras aplicaciones, en este caso el web-service de Angular.

5.2.4.1 Aplicación Lightning

Como se ha mencionado en el punto [4.2.2.](#), las aplicaciones Classic están actualmente en desuso, aunque todo proyecto Lightning se puede visualizar en su versión clásica tal y como se observa en la *figura 5-4*.

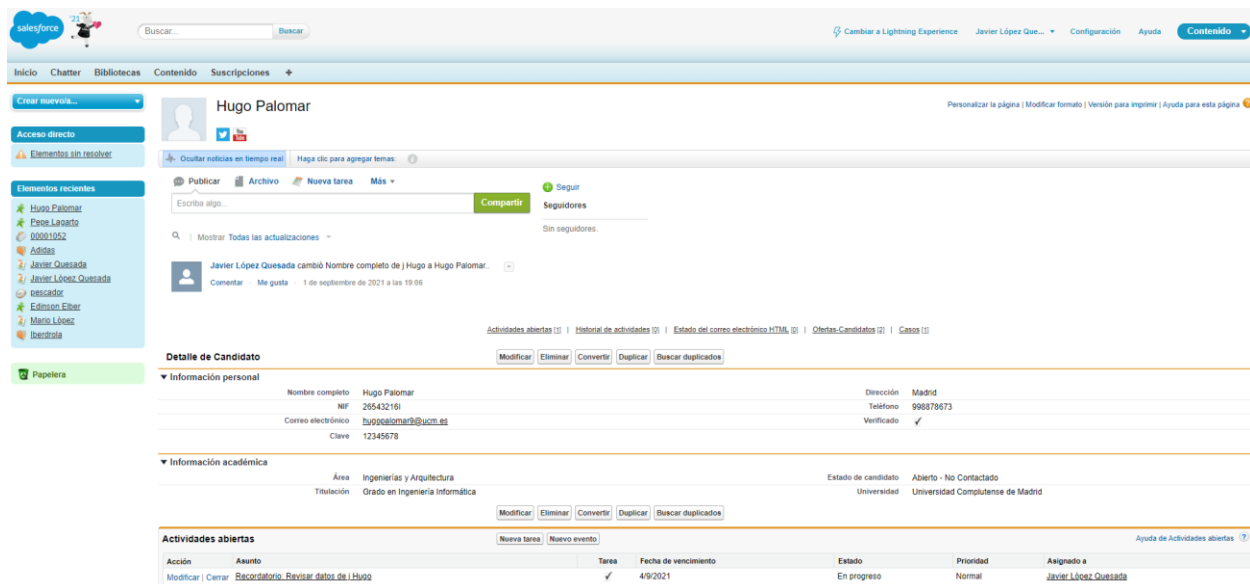


Figura 5-4. Interfaz Classic en Salesforce de PSUCM

Sin embargo, en este proyecto se ha priorizado el uso de los componentes Lightning para aportar más dinamismo a PSUCM (Figura 5-5.). Esta aplicación, cuyo nombre API es “GestorPracticasPSUCM”, es la encargada de llevar todo el apartado gráfico de la interfaz que usará el coordinador de prácticas para interactuar con Salesforce.

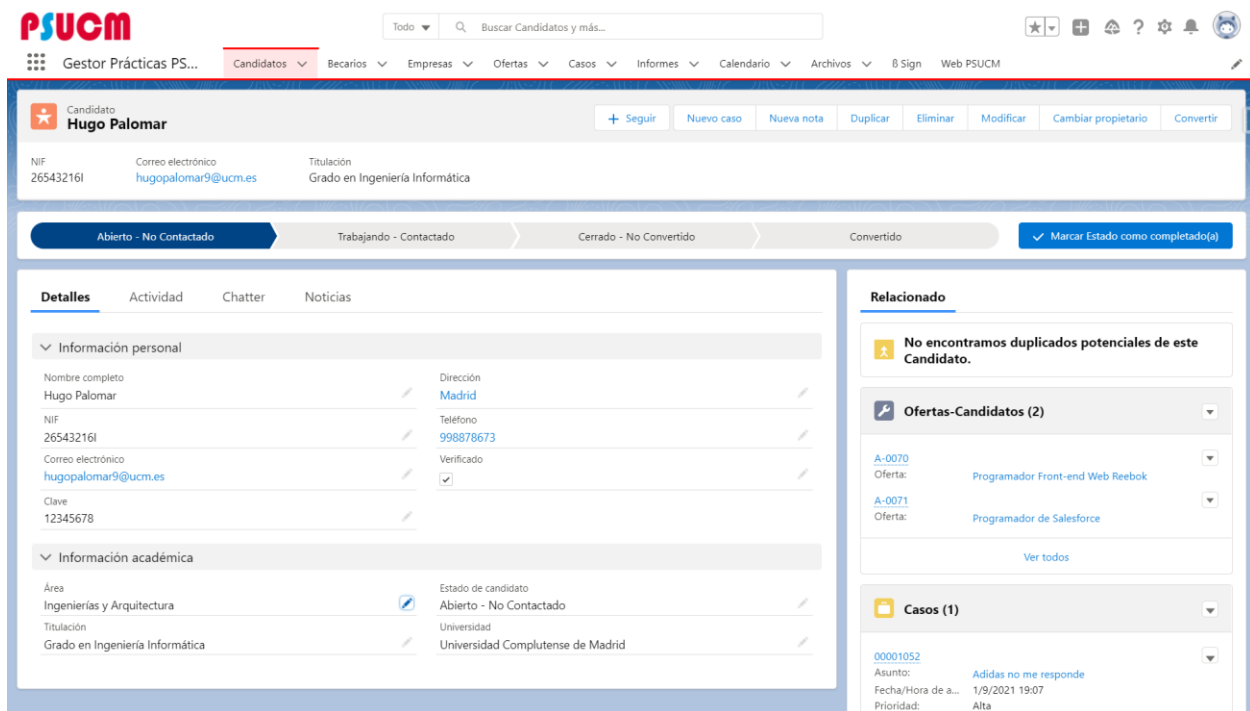


Figura 5-5. Interfaz Lightning en Salesforce de PSUCM

Analizando los elementos incluidos en la barra de navegación. Estos son:

- **Candidatos:** Objeto con el formato de página personalizado llamado "Alumno Lead" que visualiza los campos mencionados anteriormente, divididos en las secciones "Información personal" e "Información académica". Además, se han añadido las listas relacionadas de "Ofertas-Candidatos" que recoge las ofertas en las que el candidato se ha inscrito y "Casos" que alberga las incidencias que ha abierto el mismo.
- **Becarios:** Objeto con el formato de página personalizado llamado "Alumno Contact" que visualiza los campos mencionados anteriormente, divididos en las secciones "Información personal" e "Información académica". Además, se han añadido las listas relacionadas de "Ofertas" que recoge la oferta en la que ha sido elegido y "Casos" que alberga las incidencias que ha abierto el mismo.
- **Empresas:** Objeto con el formato de página personalizado llamado "Empresa Account" que visualiza los campos mencionados anteriormente, divididos en las secciones "Información empresarial" e "Información de acceso". Además, se han añadido las listas relacionadas de "Ofertas" que recoge las ofertas creadas por la empresa, "Becarios" que lista todos los becarios contratados por la empresa y "Casos" que alberga las incidencias que ha abierto el mismo.
- **Ofertas:** Objeto con el formato de página personalizado llamado "Oferta opportunity" que visualiza los campos mencionados anteriormente, divididos en las secciones "Información de la oferta" e "Detalles de la práctica". Además, se han añadido las listas relacionadas de "Ofertas-Candidatos" que recoge las ofertas en las que el candidato se ha inscrito y "Becarios elegidos" que alberga las incidencias que ha abierto el mismo.
- **Casos:** Objeto con el formato de página personalizado llamado "caso" que visualiza los campos mencionados anteriormente.
- **Archivos:** Contenedor de archivos con las carpetas "De mi propiedad", "Compartido conmigo", "Reciente", "Siguiendo" (permite ver los archivos adjuntos de los registros marcados como seguidos) y "Bibliotecas".

- **Calendario:** Objeto encargado de visualizar las diferentes tareas del coordinador. Además, usa Einstein Activity Capture para vincularlo al Google Calendar del correo institucional *psrucm@gmail.com*.
- **Web PSUCM:** Ficha web que accede al web service de PSUCM realizado en Angular.

5.2.4.2 Aplicación conectada

La aplicación conectada, con nombre API "PSUCM", es la encargada de conectar Salesforce al servicio web. Al contrario que la aplicación Lightning, no se trata de una aplicación gráfica, sino que su labor está más enfocada al tratamiento de datos. Para esta aplicación se le han otorgado todos los permisos de OAuth, que son:

- Acceder a su información básica (id, profile, email, address, phone)
- Acceder a sus datos y gestionarlos (api)
- Proporcionar acceso a sus datos a través de Internet (web)
- Acceso completo (full)
- Acceder y gestionar sus datos de Chatter (chatter_api)
- Proporcionar acceso a aplicaciones personalizadas (visualforce)
- Realizar solicitudes en su nombre en cualquier momento (refresh_token, offline_access)
- Permitir el acceso a su identificador exclusivo (openid)
- Permisos personalizados de acceso (custom_permissions)
- Acceder y gestionar sus datos de Wave (wave_api)
- Acceder y gestionar sus datos de Eclair (eclair_api)
- Acceder a servicios de Pardot (pardot_api)
- Permitir acceso a aplicaciones Lightning (lightning)
- Permitir acceso a recursos de contenido (content)
- Acceda y gestione sus datos de API de introducción de Salesforce CDP (cdp_ingest_api)

En caso de que se le quiera retirar algún permiso al servicio web se podría hacer sin problema. En cuanto a la forma de autenticación de la aplicación conectada con el servicio web se configuró de dos formas.

- **Usuario y contraseña:** En este proceso se envía un POST con un body formado por el client id y client secret de la aplicación conectada, el username y el password reforzado con un token de seguridad, mediante una llamada a `https://login.salesforce.com/services/oauth2/token`. El principal problema que presenta es que si esta llamada se hace desde el servicio web, en este caso Angular, al no estar identificado previamente salta el protocolo CORS, impidiendo realizar el POST correctamente. Pese al uso de headers como Access-Control-Allow-Origin, Salesforce no permite esquivar este protocolo desde estas aplicaciones web. Sin embargo, sí que se puede evitar este protocolo generando el token de acceso manualmente con Postman, ya que este no presenta colisiones con CORS. Por lo que bastaría con realizar la llamada con Postman y copiar el token generado al código de la aplicación.
- **OAuth 2.0 JWT Bearer Flow:** En este proceso se envía un JSON Web Token a Salesforce a través de un POST. Esto permite generar un token mediante el uso de certificados, en lugar de tener que introducir o postear las credenciales. Para generar estos certificados en Salesforce, se crea una clave privada exportable .jks generando a partir de ella un fichero .p12 que da origen a la clave privada final .key. Una vez creada esta clave y la aplicación conectada se genera el JWT codificado en base64url a través de `https://jwt.io/`. El principal problema que presenta este método es que, si se cambia de dispositivo, era muy incómodo ya que se tenían que generar las distintas claves por equipo y arrastra los problemas de CORS ya mencionados. Por otro lado, es el más seguro porque no expones tu contraseña directamente en el código. Hay un tutorial de implementación en el [\[Apéndice B\]](#).

En conclusión, debido a que se trata de un prototipo, el método de autenticación no es de vital importancia y el proyecto no tiene una alta disponibilidad económica, se realizará manualmente con Postman, añadiendo una variable global con el token de acceso, que irá variando en base a la caducidad del mismo.

5.2.5 Automatizaciones de PSUCM

En el apartado [4.1.](#) se puede observar la gran variedad de herramientas de automatización que proporciona Salesforce. Estas combinan el desarrollo de código puro con elementos de programación visual que facilitan la tarea. Como se ha mencionado anteriormente, uno de los objetivos de este proyecto es que el gestor de prácticas sea capaz de realizar cambios en objetos y automatizaciones. Es por esto que la idea es usar exclusivamente las funcionalidades que proporcionan más opciones y permitan localizar rápidamente los cambios a realizar. Siguiendo este patrón, se intenta evitar el código puro y otras herramientas algo obsoletas y limitadas como los process builder, centrándonos principalmente en flows y reglas de validación. A continuación, se listan todos los procesos no estándar creados para el proyecto PSUCM:

- **CorreoBienvenidaCandidato:** Flow que envía un correo electrónico de bienvenida a los nuevos usuarios candidatos a becarios. Además, cuando un candidato es creado, tanto desde el website como manualmente, se crea una tarea de alerta (para revisar datos del candidato) que notifica, en Salesforce, al coordinador de prácticas 24 horas después de su creación para que verifique los datos de la empresa y así permitirle el acceso desde la web. Para este proceso se usa la plantilla de código personalizado "Bienvenida alumno".
- **CorreoVerificacionCandidato:** Flow que envía un correo electrónico indicando cuando el usuario ha sido verificado por el coordinador. Cuando esto se produce, el candidato ya tendrá acceso a la web. Para este proceso se usa la plantilla de código personalizado "Verificado alumno".
- **CorreoBienvenidaEmpresa:** Flow que envía un correo electrónico de bienvenida a los nuevos usuarios que se registran como empresas. Además, cuando una empresa es creada, tanto desde el website como manualmente, se crea una tarea de alerta (para revisar datos de la empresa) que notifica, en Salesforce, al coordinador de prácticas 24 horas después de su creación para que verifique los datos de la empresa y así permitirle el acceso desde la web. Para este proceso se usa la plantilla de código personalizado

“Bienvenida empresa”. Además este flujo también asigna automáticamente el logo de la empresa siempre y cuando el código de la empresa esté añadido en el tipo de dato personalizado *LogoAccount* que está compuesto por el nombre de la empresa, el CIF y el link del logo.

- **CorreoVerificacionEmpresa:** Flow que envía un correo electrónico indicando cuando el usuario ha sido verificado por el coordinador. Cuando esto se produce, la empresa ya tendrá acceso a la web. Para este proceso se usa la plantilla de código personalizado “Verificado empresa”.
- **ConversionToContact:** Flow que envía un correo electrónico avisando a los candidatos que han sido convertidos en becarios, y por tanto su acuerdo de beca ha sido firmado y validado correctamente. De esta forma, ya podrían entrar en la web como becarios. Para este proceso se usa la plantilla de código personalizado “Bienvenida becario”.
- **SignatureAssignment:** Flow que asigna una firma automáticamente a un becario a una empresa en función de si se rellena el campo Company o el campo Contact.
- **FillContactIDOpportunity:** Flow que rellena el campo contactId para relacionar el becario elegido con su oferta correspondiente. Además, cambia la etapa de una oferta a “Asignada” cuando se elige un nuevo becario.
- **OpportunityLead:** Trigger que no permite que un candidato se apunte dos veces a la misma oferta.
- **ContactPDF:** Página de Visualforce que genera un PDF con un anexo de prácticas que agrupa datos de la oferta, la empresa y el alumno becario.
- **ContactControllerPDF:** Clase de apex encargada de controlar las consultas de campos, mediante queries SOQ, para rellenar la Visualforce ContactPDF.
- **CorreosUCM:** Regla de validación para candidatos y becarios que deben introducir un correo que acabe en “@ucm.es”, terminología empleada para todos los correos de alumnos en la Universidad Complutense de Madrid.
- **CaracteresClave:** Regla de validación para candidatos, becarios y empresas que obliga a introducir al menos 8 caracteres en el campo clave cuando se registran.

- **FormatoNIF:** Regla de validación para candidatos y becarios que deben introducir un NIF con el formato apropiado (8 valores numéricos y 1 letra).
- **VerificadoCambioEstado:** Regla de validación para candidatos que deben ser verificado antes de cambiar su estado.
- **NotasValidas:** Regla de validación para becarios, cuya nota tiene que ser un número entre el 0 y el 10.
- **CambioDeEtapa:** Regla de validación para ofertas que impide el cambio de etapa si no se ha asignado un becario a la misma.
- **CorreoDuplicadoContact:** Regla duplicada que impide la existencia de direcciones de correo duplicadas, entre el becario y el candidato, cuando se rellene este campo en la creación o edición del becario. Está configurada para que bloquee directamente la acción, ya que no tendría sentido que dos usuarios tuvieran el mismo mail de acceso.
- **CorreoDuplicadoLead:** Regla duplicada que impide la existencia de direcciones de correo duplicadas, entre el candidato y el becario, cuando se rellene este campo en la creación o edición del candidato. Está configurada para que bloquee directamente la acción, ya que no tendría sentido que dos usuarios tuvieran el mismo mail de acceso.

Cabe mencionar que existen algunos process builder y workflows realizados en la instancia, pero actualmente figuran como inactivos, ya que se condensaron en los distintos flows.

5.3 Web-service

Una vez descrita toda la parte interna con Salesforce, se describirá la aplicación del web-service realizado con Angular. Sobre la funcionalidad de la web viene definida en el diagrama de casos de la figura [5-2](#), y su diagrama de clases es también es muy similar al del propio Salesforce ilustrado en la figura 5-3. La diferencia es que este servicio web no tiene los objetos "Oferta-Candidato" y "Firma", debido a que la relación muchos a muchos se crea usando modelos de Oferta y de Candidato, y con la Firma se llama directamente a una Visualforce creada en Salesforce y compartida mediante un sitio web.

Como se ha mencionado en el análisis de requisitos, la finalidad de esta web es ser usada tanto por alumnos como por empresas para ver e interactuar con el CRM mediante su correcta integración. Este a su vez puede estar integrado con otros servicios o paquetes que aumentarán de manera exponencial las funcionalidades de la herramienta. Además, Angular es uno de los frameworks más flexibles a la hora de realizar integraciones que permiten un acabado muy profesional. En PSUCM, la interfaz de la figura X ha sido creada usando bootstrap, SASS, HTML5 y Typescript descritos en el punto [3.1](#).

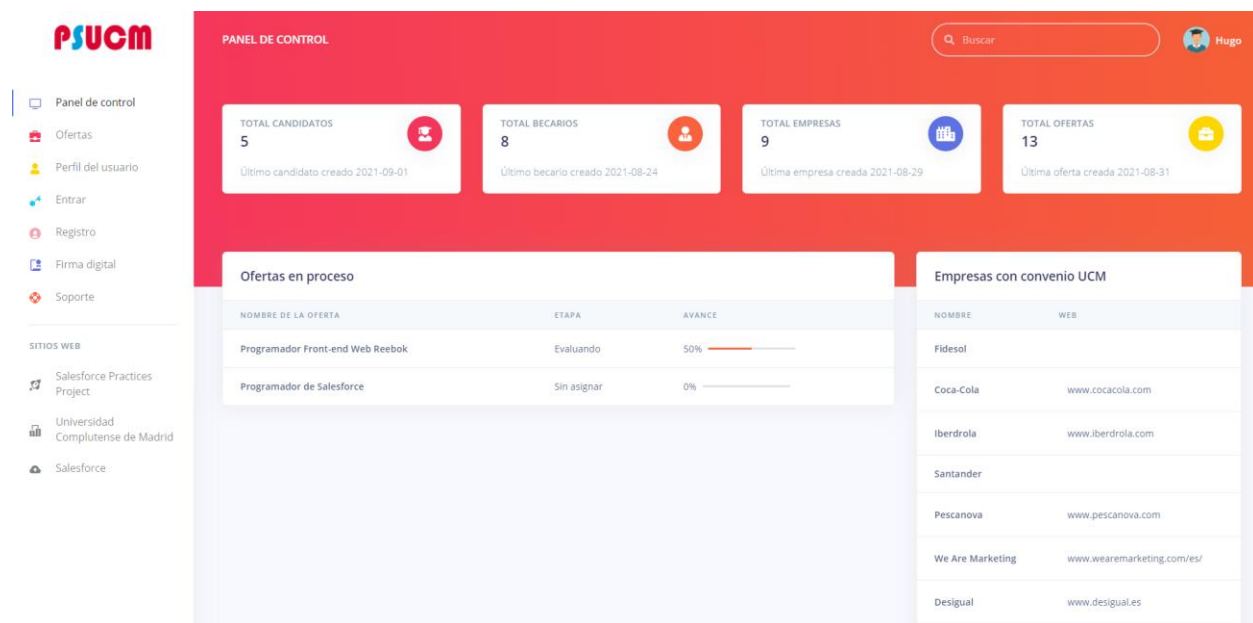


Figura 5-6. Interfaz en Angular de PSUCM

Además, esta web tiene acceso directo a la página de la Universidad Complutense, a la página principal de Salesforce y a una landing page creada en Wix para la comercialización y captación de universidades para el presente proyecto.

Analizando la landing page creada para vender el proyecto de Prácticas Salesforce (Salesforce Practices Project) se refleja de forma más clara la finalidad de una landing page. En primer lugar, la página vende el producto a través de una explicación dividida en tres puntos. En segundo lugar, aparece un apartado que plasma uno de los fines más importantes de una landing page, en el que aparece una agenda con diferentes webinars basados en Salesforce y el proyecto de Prácticas Salesforce al que el

interesado puede apuntarse de manera gratuita. De esta manera, se obtienen datos del potencial cliente como nombre, apellidos o correo electrónico, el cual puede ser usado en un futuro para recibir información, ofertas o la suscripción a una newsletter del proyecto. En tercer lugar, se muestra un apartado de universidades que hacen uso de la plataforma con sus respectivas valoraciones para así crear confianza con el interesado y por último, la opción de contactar con la empresa para cualquier tipo de duda.

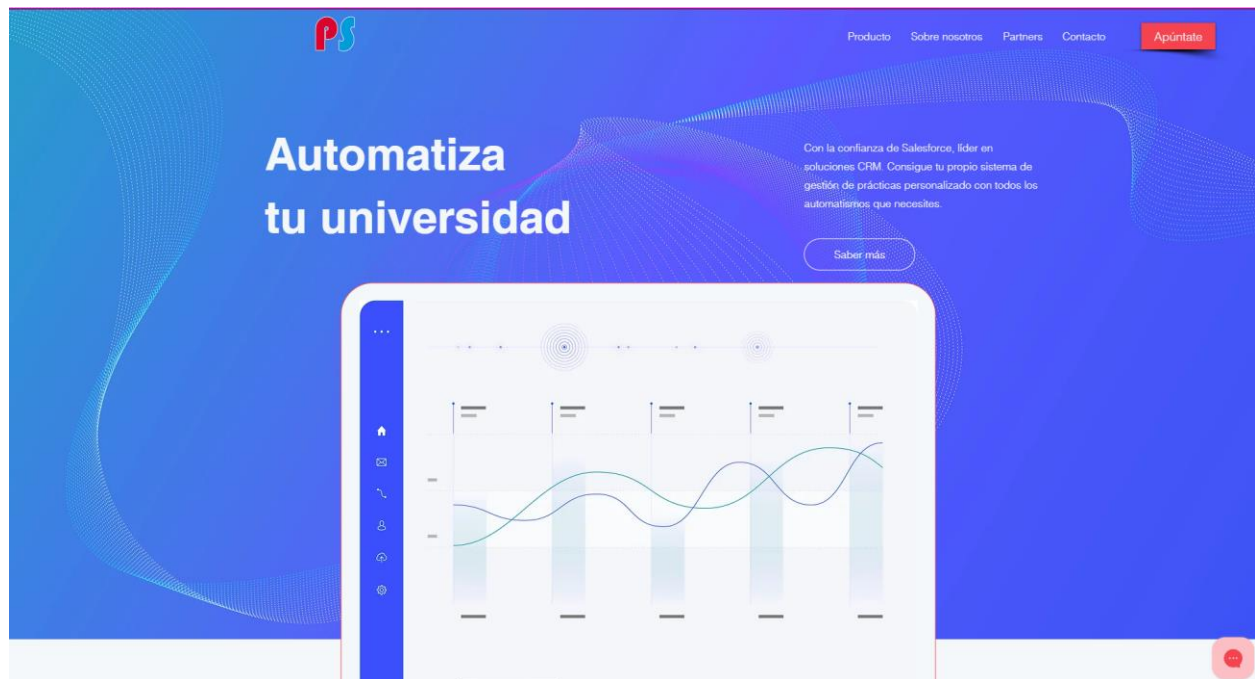


Figura 5-7. Interfaz de la Landing Page de Salesforce Practices Project

Además de estos elementos comunes en las landing pages, se ha incluido un chat para conversar con el dueño del proyecto de forma más directa.

Todos los elementos que componen el web-service dan consistencia al proyecto, alejando a los usuarios del concepto de CRM, cuando realmente lo están usando de primera mano.

5.4 Caso práctico del proceso completo

En el siguiente vídeo se incluye un ejemplo con todo el proceso, desde que se registra un alumno hasta que acaba todo el proceso de beca. Se irá navegando por Salesforce y la Web desde el punto de vista de alumno, empresa y coordinador, mostrando todas las funcionalidades citadas en el análisis de requisitos.

https://www.youtube.com/watch?v=Fb8r0VQ81VA&ab_channel=JavierL%C3%B3pezQuesada

Capítulo 6 - Conclusiones y trabajo futuro

Tras todo el análisis, la integración y el desarrollo en la propia instancia de Salesforce, se pueden sacar varias conclusiones a favor y en contra de la adaptación de un gestor de prácticas al uso estándar de Salesforce. Estas conclusiones se pueden resumir en:

- 1) Para esta adaptación, en todo momento se ha priorizado el “reciclado” o uso de objetos estándar de Salesforce, así como los procesos y automatizaciones asociados al mismo. Esto ha facilitado la implementación de algunos procesos como la conversión de alumnos candidatos a becarios, la creación de multitud de campos y restricciones o el esquema relacional inicial. Sin embargo, a medida que estos objetos van evolucionando y se crean nuevas relaciones, muchas de las funcionalidades estándar son difícilmente personalizables. Esto hace que a veces sea preferible crear un objeto personalizado antes que trabajar con objetos puramente estándar.
- 2) Una vez implementada y diseñada toda la instancia de Salesforce, la integración requiere de mucho menos esfuerzo, ya que toda la gestión de errores y generación de valores se llevan a cabo en Salesforce. Además, los cambios de paradigma en la instancia se ven muy bien reflejados en la web, cumpliendo el objetivo inicial de que el coordinador de prácticas pueda añadir requerimientos y restricciones.
- 3) El proceso definido otorga al coordinador un control total sobre el sistema, pudiendo privar a candidatos y empresas del acceso a la plataforma, ocultando ofertas, eliminando objetos e incluso bloqueando desde Salesforce el intercambio de datos total con el web-service. Esto aumenta la seguridad considerablemente, ya que el coordinador puede actuar ante errores del sistema. Por tanto, el objetivo de la seguridad también se vería cumplido.
- 4) El proceso de firma digital podría mejorarse significativamente, debido a que el paquete usado era gratuito y tiene soporte exclusivamente en inglés. No obstante, es perfectamente funcional y cubre el objetivo definido inicialmente,

siendo relativamente sencillo el cambio de paquete gestionado por uno de pago con más funcionalidad.

- 5) Pese a que Salesforce está ideado para que las empresas paguen por número de usuarios, se ha conseguido diseñar e implementar una forma muy efectiva de tener un sistema con usuarios ilimitados, mediante el uso de registros y con una cuenta Developer que sólo permite la creación de dos usuarios reales. Esto supone que se podría usar Salesforce de manera gratuita siempre y cuando no se supere el límite de llamadas y otros aspectos definidos en el punto [3.3.1](#).
- 6) Se han llevado a cabo la mayor parte de los aspectos mencionados en la entrevista del [\[Apéndice A\]](#). Por otro lado, aquello que no se ha podido implementar es debido a que suponía un gasto económico extra o a que las APIs mencionadas, como la de secretaria virtual, no es pública debido a que tiene datos sensibles.
- 7) Con respecto a otros proyectos, como los mencionados en el apartado [2.4](#), PSUCM aporta una serie de novedades:
 - a) La integración con una herramienta front-end, recurso que no se contempla en ningún otro proyecto.
 - b) Utilización de Salesforce como back-end y front-end a la vez, según el usuario que lo utilice.
 - c) Edición de la herramienta de front-end (Angular), sin necesidad de acceder al código puro. Añadiendo o modificando herramientas propias de un administrador de Salesforce.
 - d) Integración de paquetes y bots de Einstein para aportar más funcionalidades.

Con este proyecto el objetivo principal era sentar las bases de un sistema completo de gestión de prácticas aprovechando la funcionalidad que proporciona un CRM. Cumplido este objetivo, el proyecto es ampliamente escalable permitiendo añadir un sinfín de paquetes que incorpora Salesforce en su AppExchange [\[32\]](#). A continuación, se describirán y analizarán algunas de estas mejoras y cambios a futuro.

- 1) Para lanzar la aplicación al mercado, la autenticación ideal sería usando JWT [\[Apéndice B\]](#) y para solucionar los problemas de CORS se debería crear un PROXY, servidor que hace de intermediario en las peticiones de recursos. El principal problema es que el precio de mantenimiento oscila mucho según el número de peticiones que procese nuestro sistema y el presupuesto de este proyecto no permitía la obtención de este servicio.
- 2) Debido a que se trata de un prototipo, para una versión final se puede escalar a un mayor grado de complejidad en el UML de objetos. Por ejemplo, se podría introducir un objeto personalizado *Titulation__c* que aporte un nivel más en la definición de campos. Este podría incluir campos como asignaturas de la titulación, número de créditos y muchos aspectos relacionados con la titulación, ya que actualmente se ha incluido únicamente la cadena con el nombre de la titulación. Esto podría ser aplicable a otros campos como el área.
- 3) En mayo de 2019, Salesforce lanzó una plataforma blockchain que ellos mismos bautizaron como *"First Low-Code Blockchain Platform"*, o dicho de otra forma, la primera plataforma de blockchain con un enfoque de desarrollo que permite una creación más rápida, usando un modelo de programación visual. En un sistema gestor de prácticas esto sería muy útil para la realización de contratos inteligentes, ya que según el artículo sobre contratos Blockchain [\[30\]](#), esta tecnología es la forma más segura de establecer y firmar un contrato, evitando cualquier tipo de falsificación. Esto, sumado a lo que Daniel Chaver comentó en la entrevista [\[Apéndice A\]](#), haría del Blockchain un buen aliado en este tipo de sistemas. Sin embargo, aunque dispone de un aprendizaje gratuito en la web de Salesforce, la aplicación tiene un coste extra que no entraba dentro del presupuesto del proyecto.
- 4) HelloSign es una plataforma que se puede integrar con Salesforce, creando un mejor sistema de firmas digitales que también es compatible con Dropbox, Gmail, Google Docs, Oracle y Slack. Esta herramienta supera en seguridad y confianza al paquete gestionado *"Simple Signature"* explicado en el punto [3.3.6.1](#). Al igual que las herramientas citadas anteriormente, esta requiere un coste mensual adicional lo que hace que no se haya implementado finalmente.

Chapter - Introduction

This first chapter will present the motivation of the project, the objectives set at the beginning of the project and the work plan that will be carried out to achieve these objectives.

1.1 Motivation

In the midst of the coronavirus pandemic, the need for a digitization process in which our society has been improving for the last two decades has been even more emphasized. In fact, according to several CEOs of leading companies in different sectors have placed the acceleration of digitization between 5 and 10 years. For example, Nacho de Pinedo, CEO of ISDI, has stated the following: "The 60 days of confinement have accelerated the digitalization of the world by six years" [1]. This is because one of the sectors most affected by the pandemic is the business sector, since many companies are not digitized and have not been able to sell their services or products, often leading to their temporary or permanent closure.

To overcome all that has happened, many businesses have relied on digitization and digital marketing. These two concepts go hand in hand in the definition of CRM, an application that centralizes in a single database all interactions between a company and its customers. The main CRM on the market is Salesforce [2], which has an extensive list of cloud services that have evolved and grown over the years.

The union of the need for digitization, my personal discovery with Salesforce and the idea about its application in the academic environment, led me to propose the present work to improve the management platform for internships at the university, since I recently had to perform my curricular practices and I worked with the GIPE that, without detracting from the tool, remains somewhat scarce and could provide greater functionality. This project is not only applicable to the Complutense University of Madrid and could be sold to other universities through a commercial strategy under the name of "Salesforce Practices Project" [3]. It would be applicable both to universities with an obsolete manager and to universities that do not have their own practices manager, as is the case

of the University of Jaén, which currently relies on the Icaro platform to manage the implementation of practices.

1.2 Objectives

For the definition of this project it is necessary to describe a series of objectives that will be tried to be carried out in the development of the project:

- Adapt the CRM concept to a university internship management application.
- To propose two interfaces, the first one oriented towards a user with Salesforce knowledge (internship coordinator) and the second interface, outside Salesforce, that allows to be used by students and companies.
- Use Salesforce and its API Rest for the creation of the internship platform.
- Customize the platform for its different users (internship coordinator, human resources employees of each company and students opting for internships).
- Provide the internship coordinator with the ability to change validation rules or add values to the platform without the need to program directly or request it from the development team.
- Create an application that lays the foundation for the internship management process, focusing on scalability and future growth.
- Design and optimize administrative processes related to internship management.
- Improve the internship management platform that UCM currently uses (GIPE).
- Increase security in transactions with documents and information of a sensitive nature.
- Study all possible tools, which can be used for the improvement of the platform, related to the Salesforce domain. For example, Salesforce Blockchain, Hello Sign, Tableau, etc.

1.3 Work Plan

This section describes the work plan to be followed in order to achieve the objectives described in the previous section, accompanied by a Gantt chart with the foreseen planning.

1. Research on the different CRM (Customer relationship management) in the market. In this first stage, a search was made of the main CRMs, comparing them with Salesforce, to explain why this is considered the best CRM on the market, as well as the most used. In this phase we also investigated all the Salesforce functionalities, to select the most interesting ones for this project.

Research on the different APIs and integration tools with Salesforce. Once the market power of Salesforce over the rest of CRM was verified, in this second stage an evaluation of the different options for the frontend design of the application was made, as well as its integration with the different APIs available in Salesforce.

3. Adaptation of the practice management model to a CRM Salesforce application. This third stage includes the design of objects, the relationships between them and the customization of the application, in order to obtain the desired functionality.

4. Requirements specification. In this fourth stage all the functionalities of the application were defined.

5. Application design. In this fifth stage the sketches of the application were designed, both the part linked to Salesforce and the part of the chosen front-end tool (Angular).

6. Implementation. In this sixth stage we proceeded to the development of a functional prototype of an application programmed in Angular and with a backend supported by Salesforce. This application will perform the work of a job board, using techniques and concepts related to CRM and digital marketing in general.

7. Testing and final evaluation. In this last stage, a series of tests were carried out to check the correct functioning of the application, as well as to test it among a

limited group of UCM students so that they could evaluate this project with respect to the current GIPE.

Chapter - Conclusions and future work

After all the analysis, integration and development in the Salesforce instance itself, several conclusions can be drawn for and against the adaptation of a practice manager to standard Salesforce usage. These conclusions can be summarized as:

- 1) For this adaptation, at all times the "recycling" or use of standard Salesforce objects, as well as the processes and automations associated with it, has been prioritized. This has facilitated the implementation of some processes such as the conversion of candidate students to scholarship recipients, the creation of a multitude of fields and restrictions or the initial relational schema. However, as these objects evolve and new relationships are created, many of the standard functionalities are difficult to customize. This makes it sometimes preferable to create a customized object rather than working with purely standard objects.
- 2) Once the entire Salesforce instance is implemented and designed, integration requires much less effort, as all error handling and value generation is done in Salesforce. In addition, the paradigm shifts in the instance are very well reflected on the web, fulfilling the initial goal of the practice coordinator being able to add requirements and constraints.
- 3) The defined process gives the coordinator total control over the system, being able to deprive candidates and companies of access to the platform, hiding offers, deleting objects and even blocking from Salesforce the total data exchange with the web-service. This increases security considerably, as the coordinator can take action in the event of system errors. Therefore, the security objective would also be fulfilled.
- 4) The digital signature process could be significantly improved, because the package used was free of charge and is supported only in English. However, it is perfectly functional and covers the initially defined objective, being relatively easy to change the managed package for a paid one with more functionality.

5) Although Salesforce is designed for companies to pay per number of users, a very effective way to have a system with unlimited users has been designed and implemented, through the use of registrations and with a Developer account that only allows the creation of two real users. This means that Salesforce could be used for free as long as the call limit and other aspects defined in point 3.3.1 are not exceeded.

6) Most of the aspects mentioned in the interview in [Appendix A] have been carried out. On the other hand, what could not be implemented is due to the fact that it implied an extra economic expense or that the APIs mentioned, such as the virtual secretary, is not public because it has sensitive data.

7) With respect to other projects, such as those mentioned in section 2.4, PSUCM brings a number of new features:

- (a) Integration with a front-end tool, a resource that is not contemplated in any other project.

- b) Use of Salesforce as a back-end and front-end at the same time, depending on the user who uses it.

- c) Editing the front-end tool (Angular), without the need to access the pure code. Adding or modifying tools typical of a Salesforce administrator.

- d) Integration of Einstein packages and bots to provide more functionality.

With this project the main objective was to lay the foundations of a complete practice management system taking advantage of the functionality provided by a CRM. With this goal accomplished, the project is widely scalable allowing the addition of a myriad of packages that Salesforce incorporates in its AppExchange [32]. In the following, some of these enhancements and future changes will be described and discussed.

1) To launch the application to the market, the ideal authentication would be using JWT [Appendix B] and to solve the CORS problems, a PROXY should be created, a server that acts as an intermediary in the resource requests. The main problem is that the price of maintenance varies greatly depending on the number

of requests processed by our system and the budget for this project did not allow us to obtain this service.

2) Since this is a prototype, for a final version it can be scaled to a higher degree of complexity in the UML of objects. For example, a custom Titulation__c object could be introduced that provides another level of field definition. This could include fields such as degree subjects, number of credits and many aspects related to the degree, since currently only the string with the name of the degree is included. This could be applicable to other fields such as area.

3) In May 2019, Salesforce launched a blockchain platform that they dubbed the "First Low-Code Blockchain Platform", or put another way, the first blockchain platform with a development approach that allows for faster creation, using a visual programming model. In a practice management system this would be very useful for the realization of smart contracts, since according to the article on Blockchain contracts [30], this technology is the most secure way to establish and sign a contract, avoiding any kind of forgery. This, added to what Daniel Chaver commented in the interview [Appendix A], would make the Blockchain a good ally in this type of systems. However, although it has a free learning on the Salesforce website, the application has an extra cost that was not within the project budget.

4) HelloSign is a platform that can be integrated with Salesforce, creating a better digital signature system that also supports Dropbox, Gmail, Google Docs, Oracle and Slack. This tool outperforms in security and trust the managed package "Simple Signature" explained in section 3.3.6.1. Like the aforementioned tools, this one requires an additional monthly cost, which is why it was not finally implemented.

BIBLIOGRAFÍA

- [1] Fernández, S. (2020). *Los 60 días de confinamiento han acelerado seis años la digitalización del mundo*. El Confidencial. Available: https://www.elconfidencial.com/tecnologia/2020-06-11/transformacion-digital-isdi-bra_2622219/ . [Last access: 03 09 2021].
- [2] Manchar, A. and Chouhan, A. (2017). *Salesforce CRM: A new way of managing customer relationship in cloud environment*. Coimbatore, India. Available: 10.1109/ICECCT.2017.8117887 . [Last access: 03 09 2021].
- [3] López, J. (2021). *Salesforce Practices Project Landing Page*. Available: <https://prsucm.wixsite.com/practicasalesforce> . [Last access: 03 09 2021].
- [4] Patel, K., Rathod, V. R. and Prajapati, J. B. (2017). *Performance Analysis of Content Management Systems- Joomla, Drupal and WordPress*. Coimbatore, India. Available: <https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.206.3027&rep=rep1&type=pdf> . [Last access: 03 09 2021].
- [5] Annamalai, C. and Ramayah, T. (2011). *Enterprise resource planning (ERP) benefits survey of Indian manufacturing firms: An empirical analysis of SAP versus Oracle package*. Business Process Management Journal, Vol. 17 No. 3, pp. 495-509. Available: <https://doi.org/10.1108/14637151111136388> . [Last access: 03 09 2021].
- [6] FinancialForce. (2021). *FinancialForce Documentation*. Available: <https://www.financialforce.com/> . [Last access: 03 09 2021].
- [7] Minh, H., Minh, L. and Prajap and Tien, N. (2018). *An investigation into customer relationship management (CRM) market in Vietnam*. National Economic University Hanoi, Vietnam. Available: <http://www.assumptionjournal.au.edu/index.php/eJIR/article/view/4134/2439> . [Last access: 03 09 2021].
- [8] Iglesias, A. (2020). *Enrique Polo (Salesforce): "La compra de Slack va más allá de la colaboración en sí"*. El Español. Available:

- https://www.elespanol.com/invertia/disruptores-innovadores/innovadores/tecnologicas/20201223/enrique-polo-salesforce-compra-slack-alla-colaboracion/545446241_0.html . [Last access: 03 09 2021].
- [9] Gaille, B. (2018). *Microsoft Dynamics CRM Pros and Cons*. Brandon Gaille. Available: <https://brandongaille.com/14-microsoft-dynamics-crm-advantages-and-disadvantages/> . [Last access: 03 09 2021].
- [10] Gaille, B. (2019). *Adobe Experience Manager Pros and Cons*. Brandon Gaille. Available: <https://brandongaille.com/16-adobe-experience-manager-pros-and-cons/> . [Last access: 03 09 2021].
- [11] Gaille, B. (2015). *Salesforce Pros and Cons*. Brandon Gaille. Available: <https://brandongaille.com/12-salesforce-pros-and-cons/> . [Last access: 03 09 2021].
- [12] CompareCamp (2020). *Microsoft Dynamics 365 Review*. CompareCamp. Available: <https://comparecamp.com/microsoft-dynamics-365-review-pricing-pros-cons-features/>. [Last access: 03 09 2021].
- [13] CompareCamp (2020). *Adobe Experience Manager Review*. CompareCamp. Available: <https://comparecamp.com/adobe-experience-manager-review-pricing-pros-cons-features/>. [Last access: 03 09 2021].
- [14] CompareCamp (2020). *SAP CRM Review*. CompareCamp. Available: <https://comparecamp.com/sap-crm-review-pricing-pros-cons-features/>. [Last access: 03 09 2021].
- [15] CompareCamp (2020). *Oracle Sales Cloud Review*. CompareCamp. Available: <https://comparecamp.com/oracle-sales-cloud-review-pricing-pros-cons-features/>. [Last access: 03 09 2021].
- [16] CompareCamp (2020). *Salesforce Review*. CompareCamp. Available: <https://comparecamp.com/salesforce-review-pros-cons-price-leading-crm-software/>. [Last access: 03 09 2021].

- [17] Salesforce (2021). *Salesforce Help*. Salesforce. Available: <https://help.salesforce.com/s/articleView>. [Last access: 03 09 2021].
- [18] Salesforce (2021). *Salesforce Developer Documentation*. Salesforce. Available: <https://developer.salesforce.com/docs>. [Last access: 03 09 2021].
- [19] Salesforce (2021). *Salesforce Trailheads*. Salesforce. Available: <https://trailhead.salesforce.com/>. [Last access: 03 09 2021].
- [20] Jordán, J., Guijarro, M. and Hernández, J. (2018). *Feature Analysis of the "CustomerRelationship Management" Systems for Higher Education Institutions*. Institute of Technology, Marketing Department, ESIC Business and Marketing School, Valencia, Spain. Available: <https://doi.org/10.4995/muse.2018.9232>. [Last access: 03 09 2021].
- [21] Kuester, S. (2011). *Salesforce Integration in New Product Development - A Key Driver of New Product Success?*. University of Mannheim, Germany. Available: https://madoc.bib.uni-mannheim.de/29912/1/Hildesheim,_Andreas_Paper_117.pdf. [Last access: 03 09 2021].
- [22] Lokuge, S., Sedera, D., Ariyachandra, T., Kumar, S., & Ravi, V. (2020). *The Next Wave of CRM Innovation: Implications for Research, Teaching, and Practices*. College of Business and Law, RMIT University, Australia. Available: <https://aisel.aisnet.org/cgi/viewcontent.cgi?article=4200&context=cais>. [Last access: 03 09 2021].
- [23] Casado S. (2019). *Implementación de un CRM con tecnología Salesforce aplicado a la gestión de prácticas externas de la Escuela Politécnica de Cuenca*. Escuela Politécnica, Cuenca, Spain. Available: <https://ruidera.uclm.es/xmlui/bitstream/handle/10578/22044/TFG-Sara-Casado-Figueredo.pdf?sequence=1&isAllowed=y>. [Last access: 03 09 2021].
- [24] J. Hu and B. Zhang (2020), "Application of SalesForce Platform in Online Teaching in Colleges and Universities under Epidemic Situation," 2020 International

- Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE), 2020, pp. 276-279. Available: 10.1109/ICBAIE49996.2020.00065 . [Last access: 03 09 2021].
- [25] Angular. (2021). *Angular Documentation*. Available: <https://angular.io/docs>. [Last access: 03 09 2021].
- [26] López J. (2021). *gestor-practicas-ucm*. Available: <https://github.com/javilq95/gestor-practicas-ucm>. [Last access: 06 09 2021].
- [27] Creative Tim. (2019). *Argon Dashboard Angular*. Available: <https://www.creative-tim.com/product/argon-dashboard-angular> . [Last access: 03 09 2021].
- [28] Costello, L. (2018). *Advantages and Benefits of ERP Software*. Terillium. Available: <https://terillium.com/benefits-of-erp/>. [Last access: 03 09 2021].
- [29] Esquio. (2018). *Que son ERP, CRM, CMS, ECOMMERCE*. Esquio. Available: <https://esquio.es/erp-crm-cms-ecommerce/> [Last access: 03 09 2021].
- [30] Watanabe, H., Fujimura S., Nakadaira, A., Miyazaki, Y., Akutsu, A. and Kishigami, J. (2016). *Blockchain Contract: Securing a Blockchain Applied to Smart Contracts*. Muroran Institute of Technology, Muroran-City, Hokkaido, Japan. Available: 10.1109/ICCE.2016.7430693 . [Last access: 03 09 2021].
- [31] Salesforce (2021). *Salesforce Extension Pack*. Available: <https://marketplace.visualstudio.com/items?itemName=salesforce.salesforcedx-vscode>. [Last access: 03 09 2021].
- [32] Salesforce (2021). *Salesforce AppExchange*. Available: <https://appexchange.salesforce.com>. [Last access: 03 09 2021].
- [33] López J. (2021). *gestor-practicas-ucm-sf*. Available: <https://github.com/javilq95/gestor-practicas-ucm-sf>. [Last access: 06 09 2021].

APÉNDICES

Apéndice A - Entrevista a Daniel Chaver

Javier: Buenos días Daniel, en primer lugar muchas gracias por atenderme para preguntarte algunos aspectos de funcionalidad y mejora de la plataforma GIPE, así como algunos relacionados con este mismo proyecto. A continuación, te voy a hacer unas 8 preguntas y la duración aproximada será de una media hora, así que si te parece empezamos.

Daniel: Perfecto, cuando quieras.

J: Primera pregunta, ¿cuáles son las principales carencias que, según tu criterio, tiene el GIPE?

D: Con respecto a funcionalidad y funciones creo que está bien, tendría que pensar alguna cosa, pero bueno en principio no detecto ninguna carencia en cuanto a funcionalidades. En cuanto a uso a nivel de usuario si hay alguna cosa que sería mejorable pero son cosas muy específicas. Por ejemplo, no se puede acceder o vincular la información del GEA, el portal encargado de recoger los datos de la Universidad Complutense. Además, es la aplicación donde se graba la matrícula y las notas de todos los estudiantes. A veces no casan bien los datos con la información del GIPE, de tal manera que hay cosas que hay que hacer manualmente y lo ideal sería que se pudieran hacer que fueran automáticas. Por ejemplo, la información de matrícula en prácticas en nuestro caso, por algún motivo que desconozco tenemos que meterla a mano, tenemos que ir uno por uno viendo los estudiantes que están matriculados en las asignaturas de prácticas reflejando el tránsito. Es verdad que existen una funcionalidad que permite automatizar eso, pero por algún motivo nosotros no lo podemos hacer. Entonces bueno, eso es una cosa que solo hay que hacer una vez y bueno pues dentro lo que cabe tampoco es un problema grave, pero sería mejorable si se pudieran importar datos desde el GEA. Y luego otra funcionalidad relacionada con este uso, es que la formación de las prácticas de los estudiantes hay que ir la viendo siempre por titulación, no se puede ver completa. No se puede ver la información de las prácticas o

el estudiante para toda la facultad o para toda la universidad. Esto es bastante incómodo, porque si tienes unas cuantas titulaciones, nosotros tenemos seis grados, dos dobles grados y varios másteres, tienes que ir uno por uno para poder ver y modificar las prácticas o la información de los estudiantes. Y luego, con respecto a uso también hay cosas que yo ya de la costumbre pues manejo bien, pero resultan poco intuitivas al principio, cuesta un poquito encontrar la información dónde está. No sé si por los iconos o por qué o por qué hay muchas opciones diferentes en la misma ventana. Por distintos motivos, pero la conclusión es que a veces resulta poco intuitivo y seguro que si me dejaras más tiempo se me ocurrirían más cosas así, pero en principio no me voy a extender más.

J: Si, te voy preguntando y si te vienen ideas, me las comentas sin problema. Segunda pregunta, ¿destacarías alguna funcionalidad del GIPE que te facilite mucho tu labor como coordinador de prácticas?

D: En general, creo que es algo muy útil tanto porque en realidad en el GIPE se hace todo desde la publicación de la oferta, hasta la evaluación de las prácticas. Entonces toda la comunicación entre tutores académicos, coordinador de prácticas, la OPE y toda la comunicación se hace de forma muy cómoda a través del GIPE. La generación de los anexos es algo muy cómodo, ya que puedes definir el modelo que quieres como los campos que quieres que incluya y que se genera automáticamente. Todo el proceso de práctica se puede hacer dentro de la misma plataforma y a mí me resulta muy cómodo. Se hacen las empresas publican la oferta selecciona el estudiante yo desde la facultad veo la información de la gente y la puedo aprobar o cancelar o lo que sea tengo toda la información junta en el mismo sitio. Los tutores pueden hacer los informes de evaluación a través de la plataforma el estudiante todo lo que tiene que entregar lo entrega a través de la plataforma y eso hace que todo quede registrado en el mismo lugar y que luego cuando tienes que buscar información en el futuro la encuentras muy fácilmente no sé son cosas que se me va ocurriendo pero vamos en general todo me resulta cómodo. El hecho de que esté todo unificado lo veo algo que resulta muy cómodo.

J: Saliendome un poco de las preguntas que tenía preparadas, sobre este proceso que comentas, yo conozco el GIPE a nivel de alumno, pero ¿cómo es todo el proceso de gestión y qué entidades participan en él?

D: Te explico, entran tanto las empresas, como los estudiantes, como los gestores. Entonces lo primero que se hace cuando una empresa ni siquiera tiene ninguna vinculación, ni ningún acuerdo con la universidad, es crear un convenio de prácticas y eso se hace en GIPE y se transmite a través de GIPE. Una vez hecho eso, la empresa ya puede publicar prácticas en la plataforma, entonces bueno pues cuando quiere se les pasa un manual de uso de la plataforma y entonces cuando consideramos publica las ofertas: Las ofertas las publica con toda la información, con todas las condiciones de la oferta en las titulaciones a las que quiere que se abran, la modalidad si es curricular o extracurricular, bueno toda la información relacionada con la práctica. Después, si es una una práctica para la facultad de informática me llega a mí como responsable de prácticas en la facultad, mientras yo la puedo examinar porque tengo toda la información ahí y aprobarla o rechazarla. Si la apruebo, desde ese momento todos los estudiantes que estén en prácticas curriculares pueden acceder a esa oferta. Te hablo de curriculares porque las extracurriculares se gestionan desde la OPE que es el organismo responsable. Tú eso como estudiante lo conoces, entonces sabes que te puedes apuntar en las ofertas y entonces ese aviso les llegará a las empresas. Las empresas pueden, en cualquier momento, ver todas las solicitudes de realización de esa práctica que tienen y entonces ya pues realizar el proceso de selección que tengan ellos y entonces puedes seleccionar a uno o más estudiantes y eso también lo marca el GIPE. También se cambia el estado ha seleccionado o rechazado según la decisión de la empresa. Aquellos que no rechazan, cuando cambian ese estado me llega un aviso de nuevo de que el estudiante ha sido seleccionado y si todo está bien, si se cumple la normativa de prácticas entonces tramitamos los anexos que también enviamos. Bueno, esto lo enviamos a través de correo electrónico, pero sí tramita los anexos también GIPE, porque ya te digo que tenemos definido un modelo y entonces se genera automáticamente el anexo con toda la información de la práctica de las personas que intervienen en la práctica, y unos días después, antes del comienzo de la práctica, se nos devuelve el anexo firmado que también archivamos en GIPE. Por último, la práctica

va realizándose y también se rellenan en GIPE una serie de informes de evaluación. Cuando termina el estudiante, sube la memoria y realiza una encuesta. El tutor de la entidad sube la evaluación del estudiante, y el tutor académico y yo mismo, evaluamos la práctica también en GIPE, reflejando la nota en la propia plataforma y bueno ahí terminaría entonces ya todo el proceso.

J: Perfecto, pues retomamos el carrusel de preguntas y te lanzo la tercera, ¿Qué nueva tecnología o herramienta incluirías en una plataforma para la gestión de prácticas? Por ejemplo, alguna de inteligencia artificial o blockchain.

D: No sé, no se me ocurre qué se podría incluir. A ver, Blockchain supongo que estaría muy bien para hacer un control de los contratos o de los anexos y tal, podría tener sentido. En cuanto a inteligencia artificial, no lo creo porque si son funciones más estándar, entiendo que no sería necesario. Si acaso a lo mejor alguna herramienta de firma electrónica. Es decir, integrar una firma ya que hay cosas en la UCM, en las que se puede firmar. A lo mejor digo una barbaridad pero hay cosas en las que simplemente con el usuario y la contraseña se dan cosas por firmadas. Por ejemplo se está utilizando ahora en los programas de movilidad para la firma de learning agreement, hasta ahora era necesario firmar a mano en una tablet o un móvil y en la nueva versión según nos contaron hace poco simplemente con el usuario y contraseña de la ucm se considera firmado. Bueno, algo así sería cómodo en GIPE, ya que todavía seguimos arrastrando firmas escaneadas. A lo mejor eso se podría mejorar, como te decía antes es una cosa que también podría pensar con calma y seguro que se me ocurría alguna otra cosa.

J: Vale, entiendo que las firmas y los contratos son mejorables. Seguimos con la cuarta pregunta, ¿consideras el GIPE una herramienta que fomenta las automatizaciones o es plenamente manual?

D: Hay bastante correo automático y hay bastantes avisos, incluso diría que a veces demasiados porque acaba convirtiéndose en spam. Tendría también que pensarlo con más calma, no sé si hay más procesos de los que ya están automatizados que se podrían automatizar. Ya te digo que hay cosas que incluso están más automatizadas de la cuenta. Por ejemplo en mi correo, a veces me llegan avisos que ignoro porque ya me los conozco y no me dan información útil, hay algunos que sí. También, hay un proceso

que no es automatizable que es el de validar la práctica. Luego, la creación del convenio ya está está todo lo automatizado que puede estar. Yo creo que la publicación de ofertas lo mismo, o sea la publicación de ofertas requiere que primero la empresa ponga las condiciones, que bueno pues es una aplicación normal de ir rellenando campos y ya está, no hay nada que automatizar ahí. Luego, la revisión por parte de los responsables de prácticas sea lo que sea, pues requiere una supervisión de alguien que conozca las características de las prácticas que tienen que darse en las titulaciones y tal, y no veo tampoco nada que automatizar. Por último, la generación de informes está automatizada, cuando se crea la práctica y se asigna un estudiante, los informes se autogeneran. Entonces esa parte también está automatizada, hasta lo que es posible automatizar yo creo. Y nada luego es la cumplimentación de los informes y la evaluación que hacen los tutores o el estudiante. Realmente no se me ocurre nada que sea automatizable, que no lo esté ya.

J: Entiendo. Pues la quinta pregunta es, ¿tienes la capacidad de cambiar elementos de la interfaz del GIPE o establecer nuevos criterios en la creación de nuevos campos?

D: No, en principio no lo puedes configurar. Hay ciertos parámetros que puedes configurar al principio de curso. Configuras ciertas características del curso, pero no sé si eso entraría dentro de lo que preguntas, pero luego configurar el uso de la herramienta no se puede configurar, al menos que yo sepa.

J: Vale pues enlazando esto último con la sexta pregunta, ¿Tendrías algún problema en aprender unas nociones básicas de Salesforce CRM con ayuda de un manual, si así se pudieran mejorar tanto los problemas mencionados cómo gestionar mejor la propia plataforma? En caso de que la respuesta sea sí, ¿preferirías modificar elementos de configuración y programación visual o programación pura?

D: Yo te diría que no porque ya llevo mucho tiempo trabajando con esta herramienta y el hecho de aprender una nueva siempre supone un extra de dificultad, pero si no la conociera posiblemente sí. De todas formas tampoco me molesta mucho el tema del cambio de configuración. Y en cuanto a la pregunta de programación pura o visual, nosotros somos la facultad de informática pero muchas otras facultades en las que no tienen ni noción de programar entonces tendrían que ser aspectos muy básicos para

que un usuario sin conocimientos de informática pudiera utilizarlas. A veces ya cuesta que alguien utilice una herramienta si encima pretendes que pueda configurar algunas cosas, eso puede ser muy complejo para un usuario no experto, pero si viene bien detallado en un manual sería acostumbrarse y puede ser interesante. En conclusión, si tienes la opción de configurar cosas puede llegar a ser útil pero tiene que ser muy sencillo y estar muy claramente explicado, y aún así tengo mis dudas porque a lo mejor lo que a nosotros nos parece, como informáticos, algo sencillo, a un usuario no experto no le parece tan sencillo aunque sea muy visual y se lo des muy claramente explicado.

J: Vale, y para cerrar la entrevista, ¿Qué problemas han reportado las empresas o los alumnos con el sistema actual?

D: Hombre, siempre hay problemas. Aunque no lo resolvemos directamente nosotros sino que hay un mail de soporte. Estos nos ofrecen soporte cuando hay incidencias, sobre todo las incidencias críticas que hay con el uso de la aplicación, pero bueno son problemas puntuales que normalmente solucionan y que no recuerdo. Yo he escrito varias veces al mail del soporte GIPE, con problemas puntuales porque al cambiar un tutor resulta que no se generan los informes, cosas de ese estilo que puede haber en la aplicación o cosas que yo no sé y entonces necesito ayuda de un experto de uso de la aplicación. Pero cosas así gordas no se me ocurre ninguna, problemas que haya habido así de la herramienta, ya te digo yo estoy muy contento de cómo funciona, aunque es cierto que he escuchado críticas. Osea, he oído a veces críticas de que no es intuitivo, de que es muy complejo y cosas así, pero yo lo utilizo todos los días varias veces, entonces al final me he acabado haciendo experto. Al principio también tenía mis dificultades y había cosas que tampoco eran muy intuitivas, pero vamos no te sabría decir un problema gordo que se me ocurra, más bien esos problemas puntuales que nos solventan desde soporte GIPE. También entiendo que a nivel de coste y recursos económicos, supongo que una universidad con menos recursos no puede tener a todo un equipo detrás y sería interesante que el gestor de prácticas los resolviera.

Apéndice B - Manual de configuración OAuth

2.0 JWT Bearer Flow

1.- Introducción

En este tipo de flujo, el cliente envía un JWT a Salesforce a través de un POST. Salesforce lo procesa y devuelve un token de acceso que será utilizado para realizar el resto de las llamadas. Este flujo no requiere la interacción explícita del usuario y utiliza un certificado para firmar el JWT. Los principales pasos a seguir para configurar este flujo son:

- Generación de un certificado, que será usado por el cliente para firmar el JWT y por Salesforce para validarlo.
- Creación de la aplicación conectada.
- Generación del JWT por el cliente y su envío a Salesforce.

2.- Creación de certificados

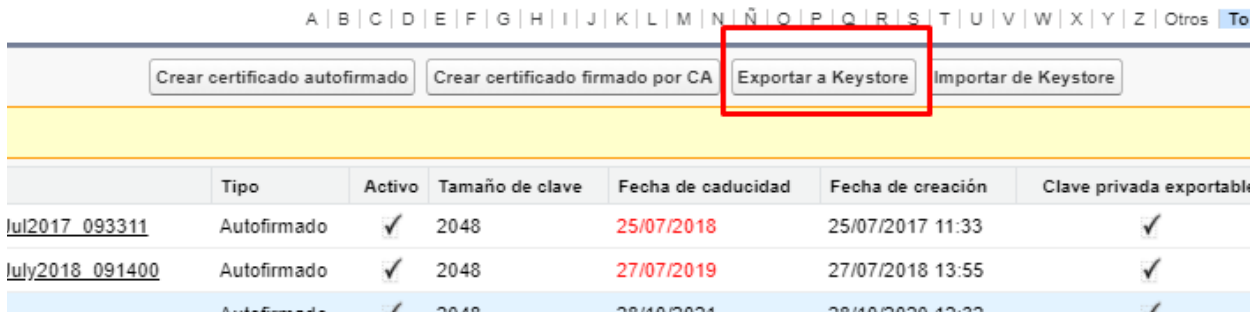
Existen dos formas de crear certificados, una es desde Salesforce y la otra es de forma manual. Independientemente de cuál sea la forma seleccionada para crear el certificado, al cliente se le debe entregar la clave privada con la que se ha generado el certificado, porque será lo que use para firmar el JWT.

2.1.- Creación desde Salesforce

Para crear el certificado en Salesforce hay que acceder a la opción "Gestión de certificados y claves" de la configuración y pulsar el botón de "Crear certificado autofirmado". Se puede seleccionar cualquier tamaño de clave y hay que seleccionar la opción de "Clave privada exportable". Una vez esté el certificado creado el siguiente paso es obtener la clave privada con la que se ha creado el certificado. Para obtener la clave privada hay que seguir los siguientes pasos:

- 1) Pulsar sobre "Exportar a Keystore".

El proxy de AJAX, algunos métodos PageReference de Apex y las llamadas HTTPS de autenticación delegada.



The screenshot shows a web interface with a navigation bar at the top containing letters A through Z and 'Otros'. Below the navigation bar is a row of four buttons: 'Crear certificado autofirmado', 'Crear certificado firmado por CA', 'Exportar a Keystore', and 'Importar de Keystore'. The 'Exportar a Keystore' button is highlighted with a red rectangular box. Below the buttons is a table with the following columns: 'Tipo', 'Activo', 'Tamaño de clave', 'Fecha de caducidad', 'Fecha de creación', and 'Clave privada exportable'. The table contains three rows of data, each representing a certificate.

	Tipo	Activo	Tamaño de clave	Fecha de caducidad	Fecha de creación	Clave privada exportable
Jul2017_093311	Autofirmado	✓	2048	25/07/2018	25/07/2017 11:33	✓
July2018_091400	Autofirmado	✓	2048	27/07/2019	27/07/2018 13:55	✓
Autofirmado	Autofirmado	✓	2048	28/10/2021	28/10/2020 12:32	✓

- 2) En la pantalla que se abre, introducir una contraseña y pulsar sobre exportar para que se descargue el fichero con extensión .jks.
- 3) Obtener el fichero .p12, que es donde está almacenada la clave privada del certificado. Para ello hay que ejecutar en la consola el comando:

```
keytool -importkeystore -srckeystore [NOMBRE_FICHERO].jks  
-destkeystore [NUEVO_NOMBRE].p12 -deststoretype PKCS12
```

Por ejemplo, si el fichero que se ha descargado se llama test.jks el comando sería el siguiente:

```
keytool -importkeystore -srckeystore test.jks  
-destkeystore test.p12 -deststoretype PKCS12
```

Pedirá introducir la contraseña que se especificó en el paso 2.

- 4) Obtener la clave privada del fichero .p12 con el siguiente comando:

```
openssl pkcs12 -in test.p12 -nocerts -nodes -out  
private.key
```

En este ejemplo, se guardará la clave privada en el fichero private.key. Este es el fichero que hay que entregar al cliente.

NOTA: Es importante ejecutar los comandos en el mismo directorio donde tenemos el .jks inicial, sino además de especificar el nombre será necesario especificar la ruta del fichero.

2.2.- Creación manual.

En este caso, no se usa Salesforce para generar el certificado, sino que se genera usando openssl. Primero se crea la clave privada y a partir de esa clave se genera el certificado. En el siguiente enlace se explican los pasos a seguir para generar el certificado usando openssl https://developer.salesforce.com/docs/atlas.en-us.sfdx_dev.meta/sfdx_dev/sfdx_dev_auth_key_and_cert.htm

3.- Creación de la aplicación conectada

Para que el cliente se integre a través de la API es necesario crear una aplicación conectada a la que se le debe asociar el certificado creado en el apartado 2 Creación de certificados.

Los pasos a seguir son:

- 1) En la configuración, acceder a "Gestión de aplicación" y pulsar sobre "Nueva aplicación conectada".
- 2) Rellenar el apartado de "Información Básica".
- 3) En el apartado de "API (Habilitar configuración de OAuth)" :
 - a. Marcar el check "Activar configuración de OAuth"
 - b. En "URL de devolución de llamada" poner la que corresponda
 - c. Marcar "Utilizar firmas digitales" y seleccionar el certificado creado en 2 Creación de certificados. Debe ser un fichero .crt. Si se ha creado desde Salesforce, se puede acceder al certificado desde "Crear certificado autofirmado" y pulsar sobre el botón "Descargar certificado".

Certificados

Ayuda para e:

Detalle de Certificado y clave

Etiqueta	Test JWT	Nombre exclusivo	testJWT
Tipo	Autofirmado	Activo	<input checked="" type="checkbox"/>
Fecha de caducidad	28/10/2021 2:00	Fecha de creación	28/10/2020 12:32
Tamaño de clave	2048	Clave privada exportable	<input checked="" type="checkbox"/> 
Certificado	C=USA, ST=CA, L=San Francisco, O=Salesforce.com, OU=00D3M0000008jjH, CN=Test Vencimiento: 28 Oct 2021 00:00:00 GMT		

d. En “Ámbitos de OAuth seleccionados” deben estar al menos seleccionadas las opciones de “Acceder a sus datos y gestionarlos” y “Realiza solicitudes en su nombre en cualquier momento (refresh_token, offline_access)”.

4) Una vez creada la aplicación conectada, en el apartado de “Políticas de OAuth” se debe poner el valor de campo “Usuarios permitidos” a “Los usuarios aprobados por el administrador se han autorizado previamente”.

5) Añadir los perfiles correspondientes en la sección de “Perfiles” de la aplicación conectada. El JWT contiene el nombre de usuario de Salesforce, por lo que el perfil que tenga ese usuario en Salesforce debe estar autorizado, sino fallará la petición.

4.- Creación de la aplicación conectada

Para generar el JWT el cliente únicamente necesita conocer el client id de la aplicación conectada y un nombre de usuario de Salesforce. En el apartado “Create a JWT” de este enlace se explica como generar el JWT:

https://help.salesforce.com/articleView?id=remoteaccess_oauth_jwt_flow.htm&type=5

Es importante tener en cuenta que todos los datos de JWT se codifican en base64url no en base64.

En el siguiente enlace hay un script en javascript que genera el JWT, únicamente hay que abrirlo y sustituir:

- <PRIVATE_KEY_FILE> por la ruta y el nombre de la clave privada
- <CLIENT_ID> por el client id de la aplicación conectada
- <USERNAME> por el nombre de usuario de Salesforce
- En caso de estar en una org de producción, sustituir:

<https://test.salesforce.com> por <https://login.salesforce.com>

5.- Petición del token

Una vez creado el JWT, se tiene que realizar un POST a Salesforce para obtener el token que se usará realizar el resto de peticiones. En el apartado "Request Access Token" de este enlace se explica cómo realizar la llamada:

https://help.salesforce.com/articleView?id=remoteaccess_oauth_jwt_flow.htm&type=5

Posibles errores

Error 500: Unable to Access Page. Este error se debe a que se ha codificado el JWT en base64 en lugar de en base64URL.

Error 400: grant type not supported. Este error se debe a que el valor en el header el grant_type es distinto a: urn:ietf:params:oauth:grant-type:jwt-bearer

Error 400: user hasn't approved this consumer. Este error se debe a que en el apartado de "Políticas de OAuth" el valor del campo "Usuarios permitidos" no está con el valor "Los usuarios aprobados por el administrador se han autorizado previamente".

Error 400: user is not admin approved to access this app. Este error se debe a que no se ha añadido el perfil correspondiente en la sección de "Perfiles" de la aplicación conectada.